AD-A041 712 STANFORD RESEARCH INST MENLO PARK CALIF F/G 17/2.1
REDUCED STATE VITERBI DETECTOR DEMONSTRATION.(U)
APR 77 S P RUSSELL, S E CANNON, A SHOHARA DCA100-76-C-0037

UNCLASSIFIED NL

1 OF 1
AD
A041712

END
DATE
FILMED
8-77

12
B.S.

Final Report *April 1977*

REDUCED STATE VITERBI DETECTOR DEMONSTRATION

By: S. P. RUSSELL, S. E. CANNON, and A. SHOHARA

Prepared for:

DEFENSE COMMUNICATIONS ENGINEERING CENTER
DEFENSE COMMUNICATIONS AGENCY
1860 WIEHLE AVENUE
RESTON, VIRGINIA 22090

CONTRACT DCA100-76-C-0037

STANFORD RESEARCH INSTITUTE
Menlo Park, California 94025 · U.S.A.

STANFORD RESEARCH INSTITUTE
Menlo Park, California 94025 · U.S.A.

Final Report *April 1977*

REDUCED STATE VITERBI DETECTOR DEMONSTRATION

By: S. P. RUSSELL, S. E. CANNON, and A. SHOHARA

Prepared for:

DEFENSE COMMUNICATIONS ENGINEERING CENTER
DEFENSE COMMUNICATIONS AGENCY
1860 WIEHLE AVENUE
RESTON, VIRGINIA 22090

CONTRACT DCA100-76-C-0037

SRI Project 4966

Approved by:

DONALD L. NIELSON, *Director*
Telecommunication Sciences Center

EARLE D. JONES, *Executive Director*
Information Science and Engineering Division

(See 1473)

Preceding Page BLANK - NOT FILMed

ABSTRACT

Early work on a new detection method applicable to band-limited satellite channels was extended. The method, termed reduced-state Viterbi detection, is a derivative of the Viterbi algorithm. In the study reported here, the potential of combining Viterbi detection of band-limited sequences with Viterbi decoding of error-correction codes was examined through simulation. A breadboard reduced-state detector was designed, and results were obtained from actual test. These results verified the predictions of the earlier study and showed that reduced-state detection can lead to significant increases in band-limited satellite channel performance.

PrecedING Page BLANK - NOT FILMed

CONTENTS

APPENDICES

Preceding Page Blank - Not Filmed

ILLUSTRATIONS

TABLES

SUMMARY

This report presents the final results of a study entitled Reduced-State Viterbi Detector Demonstration. The objectives of the study were to design and test a breadboard Viterbi detector, to examine ways in which a Viterbi detector could interface with a Viterbi decoder, and to refine some results from a prior study.

The prior study had the objective of assessing the performance and feasibility of reduced-state detection. (Appendix A gives a description of reduced-state Viterbi detection.) In accomplishing the objective of that study, fairly simple channel models were used to capture the essential parameters of the problem. The most important simplifications were that a matched-filter receiver and a linear channel were assumed. As discussed in the report of the study,[1] these simplifying assumptions were not expected to greatly affect the validity of the major conclusions. (Appendices B, C, and D present some further analysis with these simplifying assumptions relaxed.)

The results of the prior study are summarized in Figure 1, a somewhat complex plot that relates bandwidth efficiency to power efficiency for reduced-state Viterbi detection. For comparison, several other modulation schemes are included in Figure 1. Bandwidth efficiency is measured by the ratio

$$\frac{\text{bit rate (in bits/s)}}{\text{3-dB signal bandwidth (in Hz)}} \quad .$$

[1] S. P. Russell and S. E. Cannon, "Feasibility Study of Reduced-State Viterbi Detection for Capacity Improvement of Band-Limited Satellite Communication Channels," Final Report, Contract DCA100-75-C-0057, SRI Project 4375, Stanford Research Institute, Menlo Park, California (February 1976).

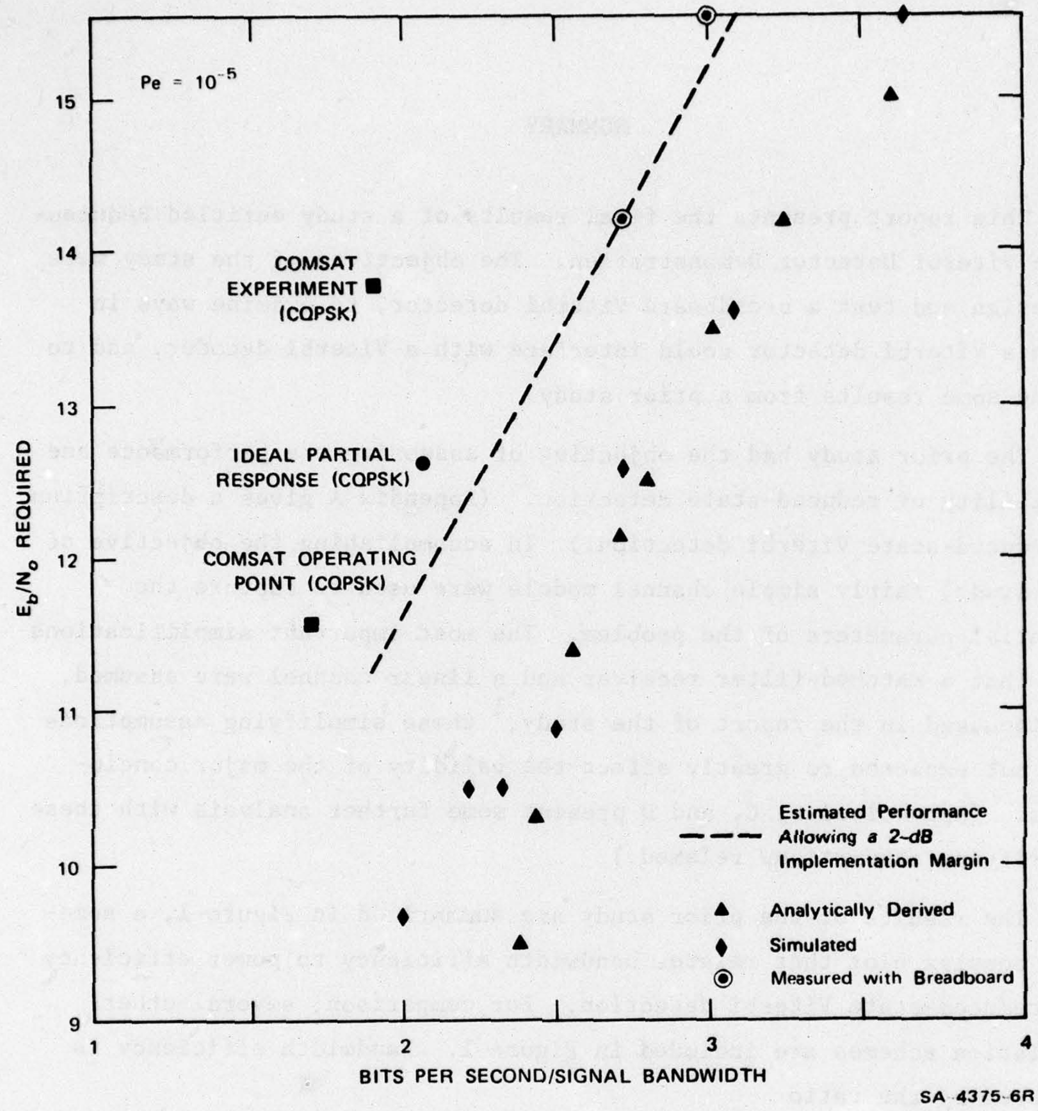


FIGURE 1 PERFORMANCE SUMMARY

Quadrature signaling is assumed, and two-sided bandwith is used. Power efficiency is measured by E_b/N_o, the energy-per-bit-to-noise-density ratio. This is proportional to the required channel signal-to-noise ratio. It is assumed that a bit-error rate of less than 10^{-5} must be achieved. Two sets of curves are shown for the performance of reduced-state Viterbi detection; one was derived analytically and the other by extrapolating data obtained from computer simulation.

These results were sufficiently promising that a decision was made to do the breadboard test, which was the major task of the work reported here. The breadboard was tested with a BPSK modem. Within the constraints imposed by the modem design, the performance predicted by Figure 1 was attained. Measured results using the breadboard are shown in Figure 1 and lie essentially on the estimated performance line. The band-limited channel throughputs equivalent to 2.7 and 3.0 bits/cycle with QPSK had degradations in required E_b/N_o to get a 10^{-5} error rate of 3.3 and 4.4 dB, respectively. The major limitation on performance at other data rates was the 3-bit uniform quantization format used by the modem.

Simulation results indicated that Viterbi decoding could be combined with reduced-state detection to achieve signaling that is more efficient than QPSK in use of both power and bandwidth.

The first section of the report gives an analysis of the feasibility of connecting a Viterbi decoder to the Viterbi detector. Following that is a brief description of the breadboard theory of operation; the report concludes with the presentation of test results.

Preceding Page Blank - Not Filmed

I INTERFACING A VITERBI DETECTOR WITH A DECODER: ARTIFICIAL SOFT DECISIONS

A. General

Both Viterbi decoding and Viterbi detection are very powerful digital communication techniques. Viterbi decoding is useful in channels that are "power hungry" but have excess bandwidth; Viterbi detection is useful in channels that are "bandwidth hungry" but have excess power. If Viterbi decoding is used in a channel that has insufficient bandwidth, the intersymbol interference caused by the bandwidth expansion factor of the code will generally cause more performance degradation than the code can regain, because the standard decoders do not take account of the intersymbol interference.

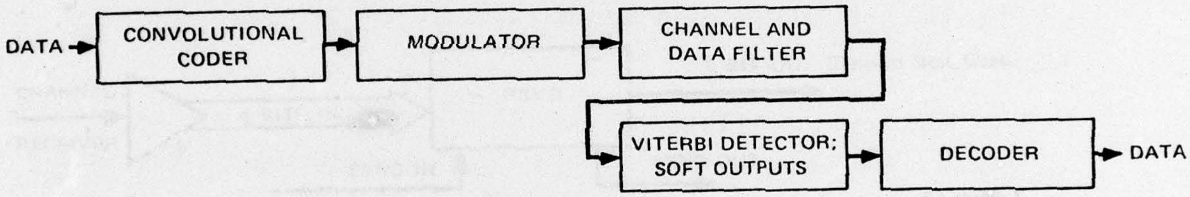
It has been known for some time that a device based on the Viterbi algorithm could be used to optimally decode convolutional codes in the presence of intersymbol interference by taking account of that interference. With such a device, one could operate in the middle ground that now divides the application areas for Viterbi decoding and Viterbi detection, realizing some of the advantages of each. Such a device, however, could be quite complicated and would not use equipment that is already in the field. We present here a method for circumventing these difficulties.

A Viterbi detector receives as input quantized data from the communication channels and uses knowledge of the channel pulse response to transform these data into guesses of the symbols that were actually put into the channel. These estimates, which are "hard" decisions, are output.

In principle, the hard outputs can be fed into an error-correction decoder, such as a Viterbi decoder. As noted above, however, the intersymbol interference caused by the bandwidth expansion of the code will result in more performance degradation than the decoder can win back.

Suppose however, that the Viterbi detector does not output hard guesses of the input bit sequences; rather, suppose it outputs 3-bit quantized data that a Viterbi decoder can operate on as if the data came out of a memoryless channel. That is, let us imagine that the Viterbi detector puts out statistics on bit probabilities that appear to be soft decisions to a Viterbi decoder.

The essence of the idea is to construct a Viterbi detector in such a way that it makes a reasonably accurate estimate of the probability that a given channel symbol is a 0 (or a 1). The estimate of this probability is then transformed into a 3-bit code that is suitable for input to a standard soft-decision Viterbi decoder. Thus, the following cascade of devices is proposed:



Details of the performance of the idea are now discussed. In the normal reduced-state algorithm, the detector or decoder is provided with an amount of path length memory equal to several constraint lengths. This memory minimizes the probability of sequence error and effectively "hardens" the output decisions by causing the final bits of all states to be equal with high probability. To output soft decisions, this extra memory in the detector is discarded. If the number of pulses that deliver 90% of the energy in the channel pulse response is N, then the total path memory used may have length as short as N. The state-tracking algorithm followed is the normal reduced-state Viterbi algorithm. However, rather than the final bit of a state being put out as a hard decision, a probability estimate that a given channel symbol is a 1 or a 0 is used to generate a soft decision. Use is made of a variety of information to estimate the probability that a given channel symbol is 1 or 0. Some of the types of information that may be used are:

- All the metrics of the retained states.
- A certain subset of these matrices. (For example, use may be made of the metric corresponding to the first sequence that differs from the most likely sequence in its estimation of the relevant channel symbol.)

B. A Proposed Implementation

Given the above information, the most straightforward way of calculating the desired probability is to apply Baye's rule to the sequences and their associated metrics. Formally, an estimate of the probability that a 1 was sent in the i^{th} position is:

$$\frac{\displaystyle\sum_{\substack{\text{metrics estimating that}\\ \text{a 1 was sent in the } i^{th}\\ \text{position}}} e^{-\text{ metric}}}{\displaystyle\sum_{\substack{\text{metrics estimating that}\\ \text{a 1 was sent in the } i^{th}\\ \text{position}}} e^{-\text{ metric}} + \sum_{\substack{\text{metrics estimating that}\\ \text{a 0 was sent in the } i^{th}\\ \text{position}}} e^{-\text{ metric}}},$$

where the metrics are appropriately scaled. Because of the hardening effect of merging candidate paths in the past, i is chosen to be just large enough that most of the energy due to the i^{th} bit has been received. For example, if the channel pulse response is

$$\ldots\ 0.3,\ 0.85,\ 0.35,\ -0.2,\ 0.1,\ \ldots,$$

then i may be 3 or 4.

This expression may be considerably simplified by assuming that only two metrics will materially affect the result. The two are the smallest metric estimating that a 0 was sent and the smallest metric estimating that a 1 was sent. Let us call these metrics M_0 and M_1, respectively. Then the probability estimate $\hat{P}_1$ that a 1 was sent, given this information, is

$$\hat{P}_1 = \frac{e^{-M_1}}{e^{-M_1} + e^{-M_0}} \quad \text{or}$$

$$\hat{P}_1 = \frac{1}{1 + e^{M_1 - M_0}} \quad .$$

Here $\hat{P}_1$ will be used to generate so-called artificial soft decisions for input to a decoder. The objective of the artifical soft decisions is to imitate a memoryless channel for the decoder. The transformation from $\hat{P}_1$ to 3-bit soft decisions can be heuristically derived as follows.

If a 1 is sent over a memoryless additive white-Gaussian-noise channel, the probability density of the output X is

$$f(X|1) \; \alpha \; e^{-(X-m)^2/2\sigma^2}$$

where M is the expected value of the output if a 1 is sent; σ^2 is the channel noise variance; and X is the actual channel output. By Bayes' rule, the probability that a 1 was sent over the memoryless channel, given that X is received, is

$$P_1(X) = \frac{f(X|1)}{f(X|0) + f(X|1)} \quad .$$

Since we are dealing with binary antipodal signaling,

$$P_1(X) = \frac{e^{-(X-m)^2/2\sigma^2}}{e^{-(X-m)^2/2\sigma^2} + e^{-(X+m)^2/2\sigma^2}} \quad .$$

This reduces to

$$P_1(X) = \frac{1}{1 + e^{-2Xm/\sigma^2}} \quad .$$

To generate artificial soft decisions, we must generate an equivalent X, given $\hat{P}_1$ from the reduced-state detector. This X is found by solving

$$\hat{P}_1 = P_1(X)$$

or

$$\frac{1}{1 + e^{M_1 - M_0}} = \frac{1}{1 + e^{-2Xm/\sigma^2}} .$$

The solution is

$$X = (M_0 - M_1) \cdot \text{scale factor}$$

Thus it is fortunately very easy to implement artificial soft decisions.

C. Simulation of Artificial Soft Decisions

The method described above for calculating artificial soft decisions was simulated on a computer. A 9-pole Chebyschev band-limiting filter was assumed. The results are shown in Table 1, and are compared with output from an ideal memoryless additive white Gaussian-noise channel. The column headed "Simulated" shows the simulation results at the given E_b/N_o; the column headed "Ideal" shows the distribution that would be obtained with the ideal memoryless channel at a <u>different</u> E_b/N_o. The distributions are nearly identical. This shows that the simulated band-limited channel is <u>equivalent</u> to the memoryless channel. The difference in E_b/N_o represents the loss due to intersymbol interference.

Actually, the artificial soft-decision output will exhibit memory. That is, the outputs will be correlated. This is not true for the output of a memoryless channel. The two channels will be precisely equivalent only if the artificial soft decisions are interleaved to destroy the correlation. In fact, it is possible to take advantage of the correlation so that the artificial channel will be preferable to the

Table 1

ARTIFICIAL SOFT DECISIONS

Number of Channel Symbols	Bandwidth Efficiency: 2.5 bits/cycle	
	Simulated	Ideal
	E_b/N_o = 5.0 dB	E_b/N_o = 2.1 dB
Sent	10,119	10,000
Received		
Completely correct	6,192	6,179
One step off	1,707	1,702
Two steps off	1,212	1,151
Three steps off	628	609
Four steps off	271	252
Five steps off	88	81
Six steps off	17	21
Completely wrong	4	5
	Bandwidth Efficiency: 2.3 bits/cycle	
	E_b/N_o = 3.0 dB	E_b/N_o = 1.0 dB
Sent	10,119	10,000
Received		
Completely correct	5,352	5,319
One step off	1,800	1,871
Two steps off	1,522	1,409
Three steps off	879	831
Four steps off	383	382
Five steps off	140	139
Six steps off	35	39
Completely wrong	8	10

memoryless channel. However, if interleaving is used, the channels are equivalent.

The tabulation below shows the performance that can be obtained by cascading a reduced-state detector, an interleaver, and a rate 3/4 Viterbi decoder--at a bit error rate of 10^{-5}.

1.7 bits/cycle	at	E_b/N_o = 7.6 dB
1.9 bits/cycle	at	E_b/N_o = 8.4 dB

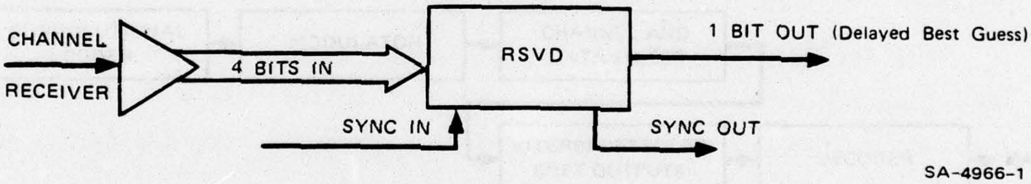
The increase in intersymbol interference caused by the rate 3/4 coder is more than compensated for by the coding gain when artificial soft decisions are used. The result is that data can be sent at higher data rates and lower E_b/N_o than are required for ordinary QPSK. By taking advantage of the correlation in the artificial soft decision output, even better performance should be achievable.

II BUILDING A REDUCED-STATE DETECTOR

A reduced-state Viterbi Detector (RSVD) was built to verify the performance projected in the earlier study.[1] The detector was built with the capability of operating with 4-bit quantized input data. However, because existing modems provide only 3-bit output data, this capability was never actually used. The theory of detector operation is now described.

A. Theory of Operation--Hard Decision

The detector receives a 4-bit channel output word as input and produces, after a delay of N such inputs, an output of 1 bit corresponding to the detector's best guess as to whether a 0 or a 1 was sent. (Here N = 24.)



To do this, the detector stores a list of guesses as to what the last N bits were supposed to be and updates this list after every new channel output is read. These guesses are simply N-bit words stored in RAM; they are called states. At the end of an update, the "best" state is determined and the last bit, that is, the oldest bit, for that state is chosen as the output bit for that cycle. Almost all the complexity comes from deciding which of the states is "best" and which of the huge number of possible states (2^N) are worth keeping.

An optimal RSVD does this by the following algorithm. For each state left from the previous cycle, two new states ("progeny") are generated by dropping the last bit in the old state and appending a 0 to the beginning for one new state, and a 1 for the other. (See Figure 2.) The predicted channel response for the supposed input sequences corresponding to these new states is then computed by using information about the channel response stored somewhere in the RSVD. The actual channel response (the 4 bits read in) is then subtracted from the predicted response for each new state, and the result is squared to form a measure, or metric, of how far off the response is from the actual response. The smaller the metric, the "closer" the corresponding guess (state) is considered to be to the correct but unknown channel input. In other words, the probability of a state's being correct is inversely related to its metric.

This does not give sufficient information to decide whether to save the corresponding state. The value of the metric for the ancestor state (the "old metric") must be added to this new measurement to yield a new metric. By doing this, the past history of the state is taken into account, and a state with consistently low metrics will do better than one that has a low measurement only for one cycle.

The RSVD algorithm picks the m closest metrics (where m = 8), saves them from the next cycle, and discards all others. As a further simplification, which only slightly degrades performance, the SRI device keeps

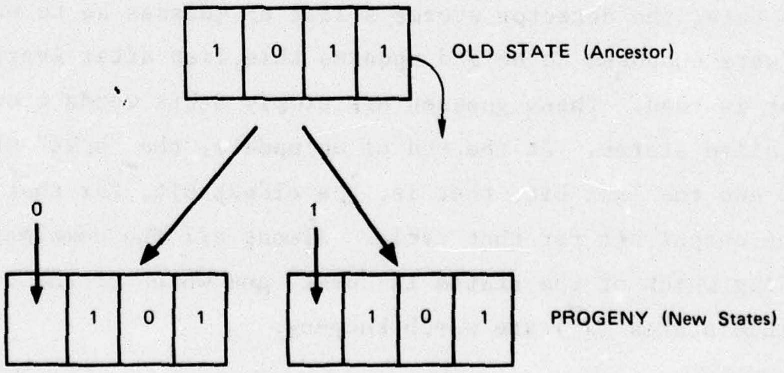
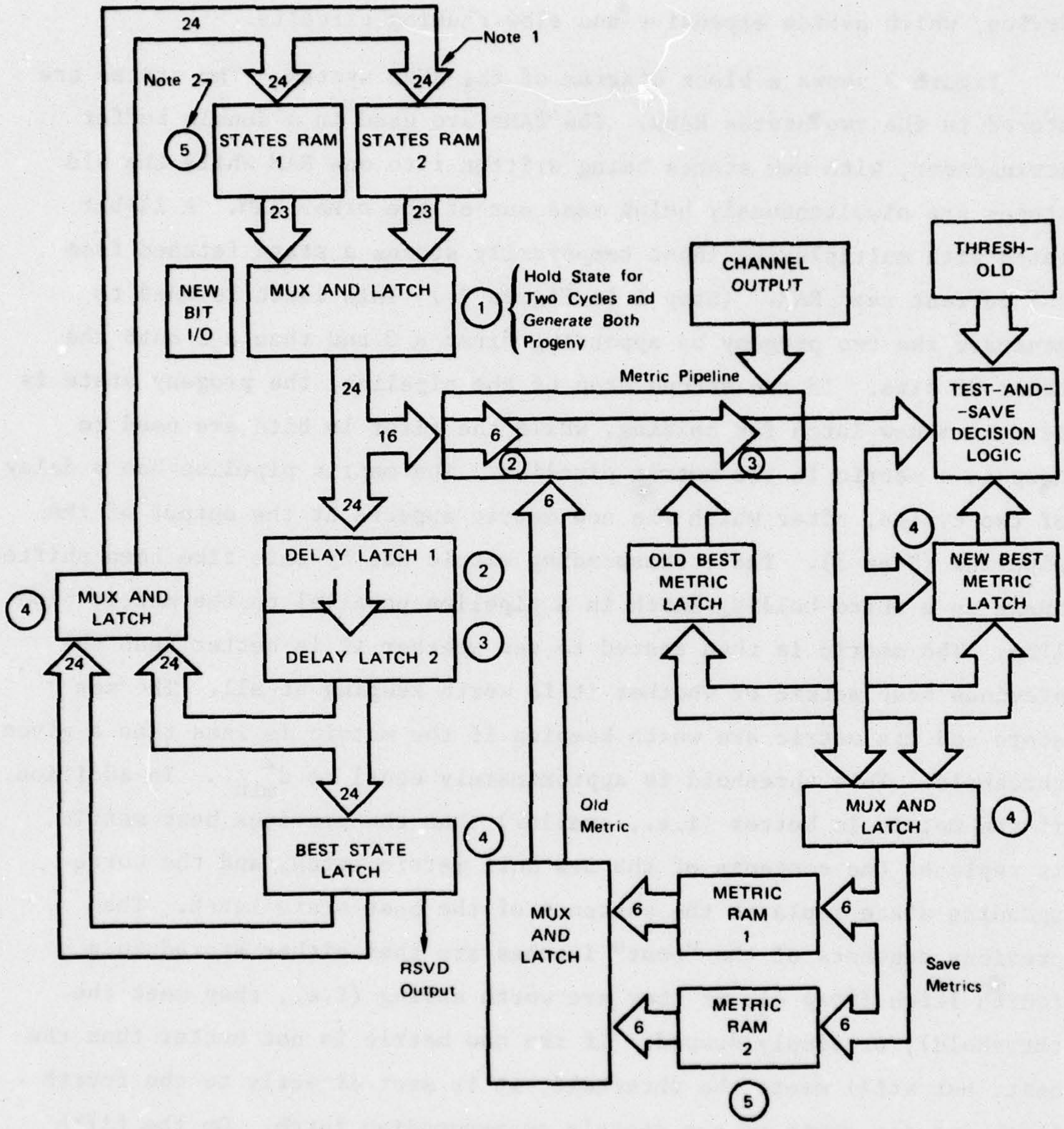


FIGURE 2 DIAGRAM SHOWING GENERATION OF NEW STATES

the best state (and its metric) and keeps up to m - 1 other states that have metrics less than a given threshold. The method for selecting these other states (described later) allows a pipeline architecture for the device, which avoids expensive and slow ranking circuits.

Figure 3 shows a block diagram of the RSVD system. The states are stored in the two states RAMs. The RAMs are used in a double buffer arrangement, with new states being written into one RAM while the old states are simultaneously being read out of the other RAM. A 24-bit latch with multiplexing input temporarily stores a state fetched from the current read RAM. (Step 1 in Figure 3.) This latch is used to generate the two progeny by appending first a 0 and then a 1 onto the other 23 bits. In the second step of the pipeline, the progeny state is sent to a new latch for holding, while the first 16 bits are used to compute a metric in the metric pipeline. The metric pipeline has a delay of two cycles, after which the new metric appears at the output of the pipeline (Step 3). The corresponding metric has by this time been shifted again to a third holding latch in a pipeline parallel to the metric pipeline. The metric is then tested to see whether it is better than the previous best metric or whether it is worth keeping at all. The new state and its metric are worth keeping if the metric is less than a given threshold. This threshold is approximately equal to d^2_{min} . In addition, if the metric is better (i.e., smaller) than the previous best metric, it replaces the contents of the new best metric latch, and the corresponding state replaces the contents of the best state latch. The previous contents of the "best" latches are then either stored in a fourth latch (Step 4), if they are worth saving (i.e., they meet the threshold), or simply dumped. If the new metric is not better than the best, but still meets the threshold, it is sent directly to the fourth latch and its state to the state's corresponding latch. On the fifth step, provided that one or the other of the save conditions has been met, the contents of the fourth latch are written into the current write RAMS for states and metrics. If the RAMS have only one space left, the writing is suppressed until all the metrics have been computed, and then



NOTE 1: Numbers in arrows indicate number of bits in parallel.

NOTE 2: Numbers in circles indicate position of data at a given step in the pipeline cycle.

SA-4966-3

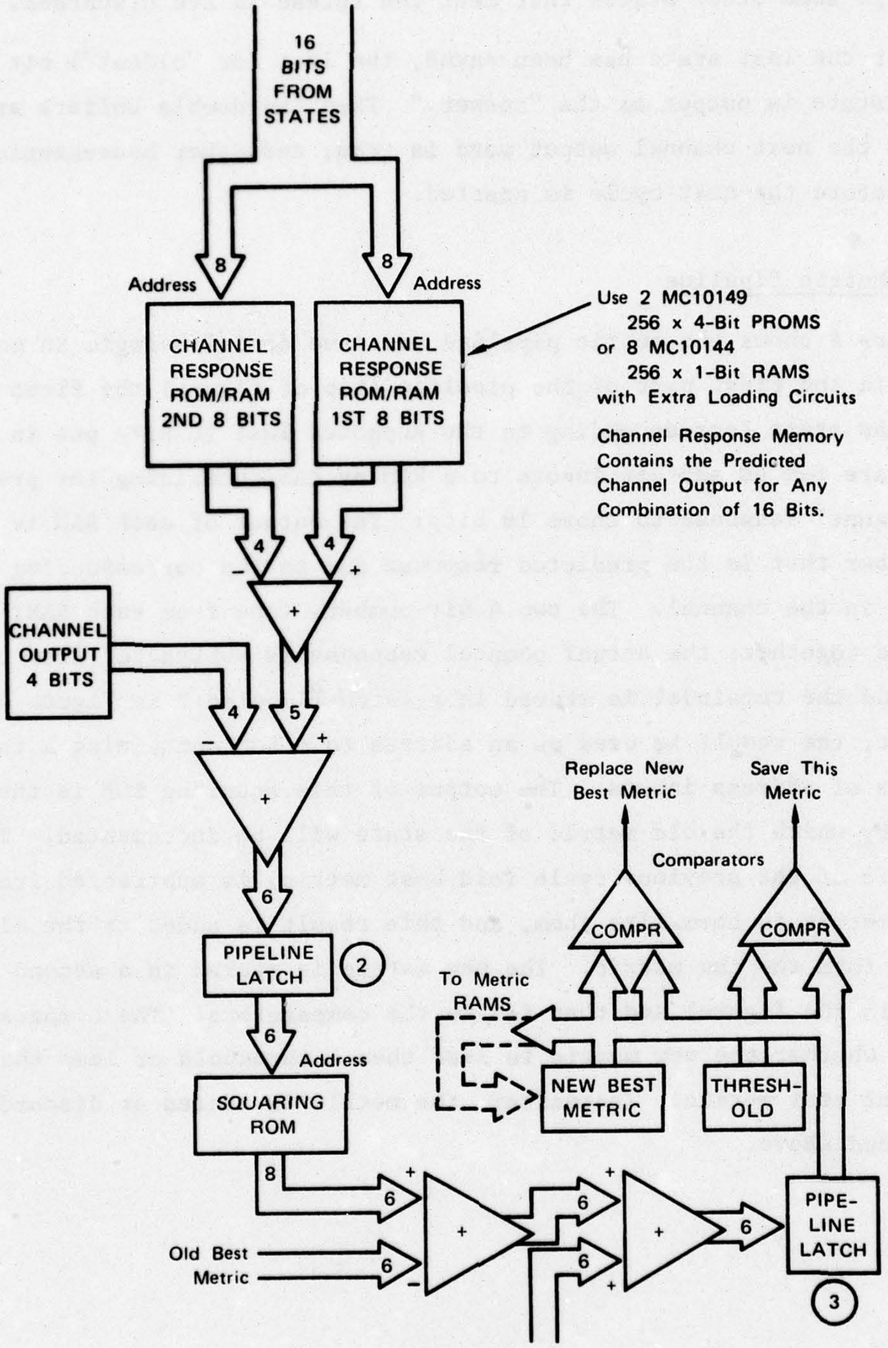
FIGURE 3 BLOCK DIAGRAM SHOWING DATA PATHS OF PEDUCED-STATE VITERBI DETECTOR

the contents of the "best" latches are written into the last addresses of the RAMS. Thus, if space runs out, the best state is still stored, even though some other states that meet the threshold are discarded.

After the last state has been saved, the last (or "oldest") bit of the best state is output as the "answer." Then the double buffers are switched; the next channel output word is read; and other housekeeping is done before the next cycle is started.

B. The Metric Pipeline

Figure 4 shows the metric pipeline and save decision logic in more detail. In the first part of the pipeline (top of figure) the first 16 bits of the state (corresponding to the supposed last 16 bits put in the channel) are fed as address inputs to a ROM or RAM containing the predicted channel response to those 16 bits. The output of each RAM is a 4-bit number that is the predicted response due to the corresponding set of 8 bits in the channel. The two 4-bit numbers (one from each RAM) are then added together; the actual channel response is subtracted from the result; and the remainder is stored in a latch (labeled 2 in Figure 4). After that, the result is used as an address to a ROM containing a table of squares of address inputs. The output of this squaring ROM is the quantity by which the old metric of the state will be incremented. The best metric of the previous cycle (old best metric) is subtracted from each old metric to normalize them, and this result is added to the old metric to form the new metric. The new metric is stored in a second latch (3 in the figure) and then fed to the comparators. The comparators determine whether the new metric is less than a threshold or less than the current best metric. Thereafter, the metric is stored or discarded, as explained above.

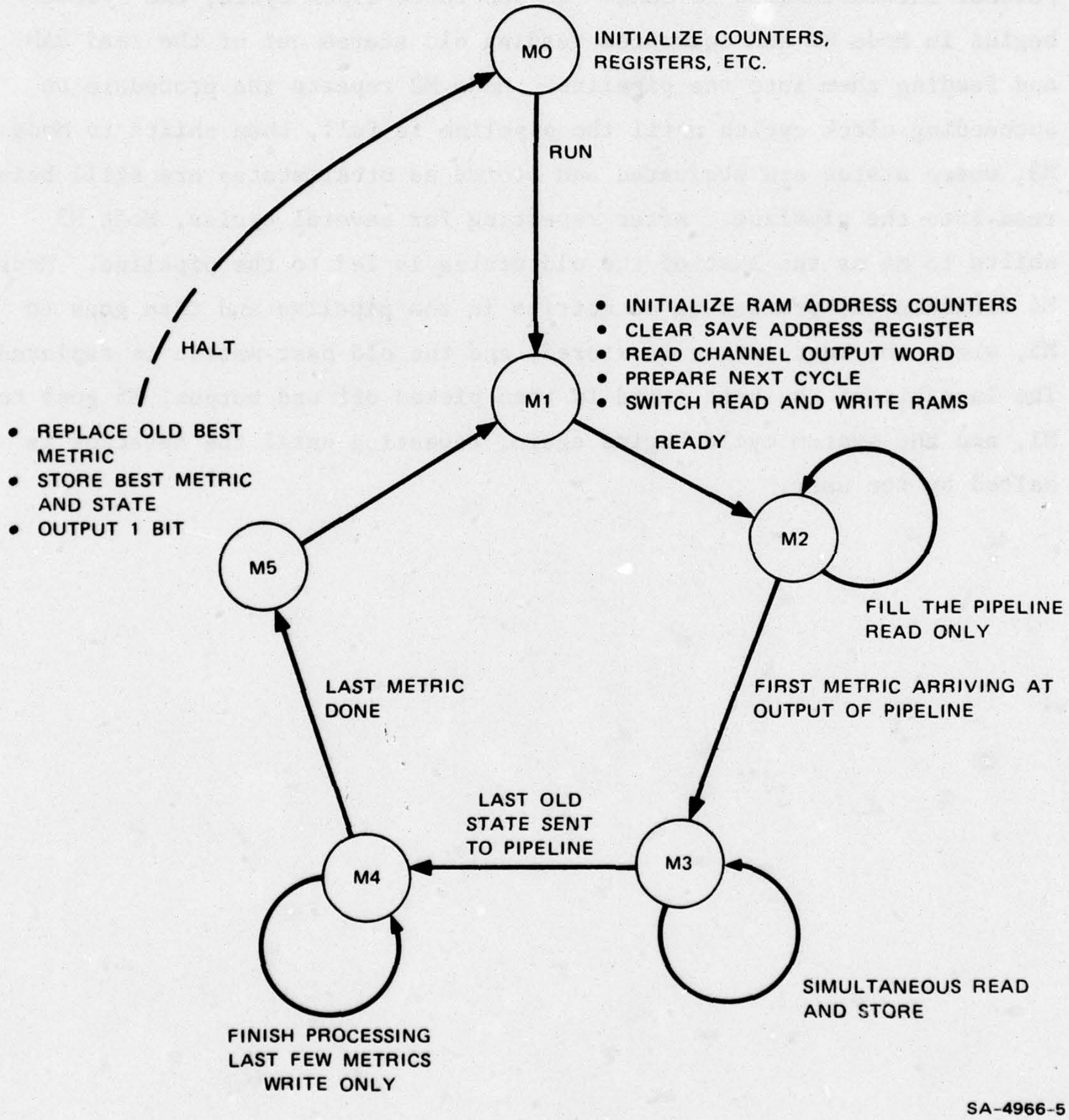


SA-4966-4

FIGURE 4 BLOCK DIAGRAM OF METRIC PIPELINE

C. System Timing and Mode Control

Figure 5 shows a diagram of the normal modes of operation of the detector. By combining a system clock divided into nine 5-ns periods with the modes control, the timing of various functions, such as read,



M0
INITIALIZE COUNTERS, REGISTERS, ETC.
RUN
HALT
• INITIALIZE RAM ADDRESS COUNTERS
• CLEAR SAVE ADDRESS REGISTER
• READ CHANNEL OUTPUT WORD
• PREPARE NEXT CYCLE
• SWITCH READ AND WRITE RAMS
M1
• REPLACE OLD BEST METRIC
• STORE BEST METRIC AND STATE
• OUTPUT 1 BIT
READY
M2
M5
FILL THE PIPELINE READ ONLY
LAST METRIC DONE
FIRST METRIC ARRIVING AT OUTPUT OF PIPELINE
LAST OLD STATE SENT TO PIPELINE
M4
M3
SIMULTANEOUS READ AND STORE
FINISH PROCESSING LAST FEW METRICS WRITE ONLY
SA-4966-5

FIGURE 5 DIAGRAM OF THE DETECTOR MODES

write, and initialize, is easily controlled. Figure 5 shows a diagram of the six modes, each of which corresponds to a particular set of tasks to be executed during a 45-ns clock cycle (nine periods) or set of cycles. Mode M0 is the start-up mode during which initialization chores are performed. After one clock cycle, the system goes to Mode M1, where further initialization is done. On the third clock cycle, the system begins in Mode M2 and commences reading old states out of the read RAM and feeding them into the pipeline. Mode M2 repeats the procedure on succeeding clock cycles until the pipeline is full, then shifts to Mode M3, where states are evaluated and stored as other states are still being read into the pipeline. After repeating for several cycles, Mode M3 shifts to M4 as the last of the old states is fed to the pipeline. Mode M4 finishes the processing of metrics in the pipeline and then goes to M5, where the best metric is stored, and the old best metric is replaced. The last bit of the best state is then picked off and output, M5 goes to M1, and the system cycle begins again, repeating until the detector is halted by the user.

III TESTING THE DETECTOR

The performance of the reduced-state Viterbi detector was tested using the procedures of the test plan given in Appendix E. Because this test plan was submitted prior to the actual tests, problems arose during the tests that had not been foreseen. In the discussion that follows, these variances are noted.

A. Modem Baseline Performance

Modem performance in the region that was not band-limited was found to be within 1 dB of theoretical. Performance in the presence of filtering was found for the following data rates: 232, 300, 340, 360, and 400 kbits/s. These rates correspond to 2.0, 2.7, 3.0, 3.2, and 3.6 bits/cycle for QPSK signaling. The results are shown in Table 2. The performance degradation at high bandwidth efficiency is dramatic, as expected.

B. Filter Response Measurement

The baseband equivalent impulse response of the 225-kHz surface-acoustic-wave (SAW) filters was measured with the aid of a double-balanced mixer operated as an amplitude modulator. A photograph of the response as displayed on an oscilloscope is given in Figure 6. It is apparent from the photograph that these are excellent filters with very little ringing.

Because of the effects of modem filtering and sampling, quantization, and automatic gain control (AGC), this simple measurement of filter impulse response did not allow us to program the channel response ROMs. We therefore proceeded to measure the channel pulse response with modem effects included, by performing a cross correlation between a PN sequence input to the channel and the channel output. (This procedure is discussed

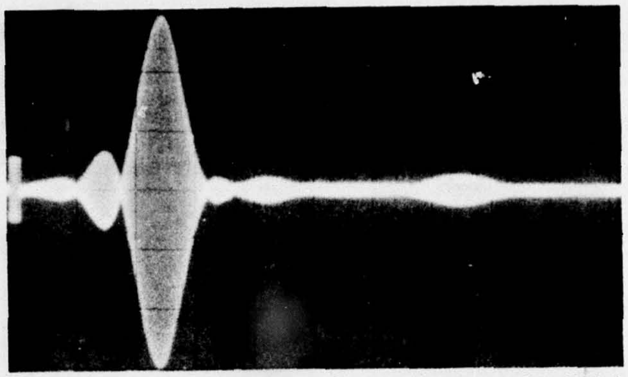
Table 2

MODEM BASELINE PERFORMANCE WITH THE 225-kHz FILTER

E_b/N_o	P_e	E_b/N_o	P_e
237 kbits/s (2.0 bits/cycle)		360 kbits/s (3.2 bits/cycle)	
9.1	10^{-2}	8.3	$5\cdot10^{-2}$
11.1	10^{-3}	12.3	$3\cdot10^{-2}$
12.1	$2\cdot10^{-4}$	16.3	$2\cdot10^{-2}$
13.1	$6\cdot10^{-5}$	22.3	$2\cdot10^{-2}$
14.1	10^{-5}	∞	10^{-2}
300 kbits/s (2.7 bits/cycle)		400 kbits/s (3.6 bits/cycle)	
7.9	10^{-2}	11.4	0.07
9.9	$6\cdot10^{-3}$	21.4	0.06
11.9	$2\cdot10^{-3}$	∞	0.06
13.9	10^{-3}	–	–
15.9	10^{-4}	–	–
17.9	$2\cdot10^{-5}$		
19.9	$2\cdot10^{-6}$		
340 kbits/s (3.0 bits/cycle)			
12.8	10^{-2}		
–	–		
16.8	$5\cdot10^{-3}$		
18.8	$3\cdot7\cdot10^{-3}$		
–	–		
∞	$2\cdot10^{-4}$		

in Appendix E.) A major problem that was to affect all future measurements was discovered in the process of performing the measurement. This problem was our inability to control the modem AGC functioning.

The modem used in the test has two AGC functions. One of them adjusts the level of the received RF signal to a standard level. The second adjusts the level of the demodulated signal prior to quantization,



SA-4966-6

FIGURE 6 FILTER IMPULSE RESPONSE—5 μs PER DIVISION

and it was this AGC function that resulted in some difficulties. The essence of the problem was that the AGC increases the level of the demodulated signal so much that most of the quantized outputs are at the extreme possible values. In other words, the intermediate quantization levels are seldom used, and there is an effective loss of level information. Because it is exactly this level information that the reduced-state detector uses to make its output decisions, this second AGC function gravely threatened the potential performance of the breadboard. The basic problem was that the AGC was not designed to operate with heavily filtered signals.

We attempted to solve the problem by inserting a variable voltage divider in the modem. The divider was inserted at the input to the quantization function. Using this voltage divider, we could exercise some control over the distribution of levels in the quantized output.

We also abandoned the cross-correlation method of pulse response measurement for a simpler and more direct method. To measure pulse response with the direct method we simply compared channel input to quantized channel input. We found that the quantized channel output was essentially determined by three channel input bits. The actual dependence extended over more than 3 channel bits at high signaling rates, but the 3-bit level quantization hid this effect from us.

Channel response patterns were generated at 300 and 400 kbits/s for various settings of the voltage divider. Four patterns that performed well in the tests are shown in Table 3. These patterns have been labeled A through D. A ROM was programmed with each pattern.

Table 3

CHANNEL RESPONSE PATTERNS

Preceding Bits	Output*	Preceding Bits	Output
A 300 kbits/s		B 300 kbits/s	
000	-4	000	-4
001	-3	001	-4
010	1	010	2
011	2	011	2
100	-3	100	-3
101	-2	101	-3
110	3	110	3
111	3	111	3
C 400 kbits/s		D 400 kbits/s	
000	+4	000	-4
001	-3	001	-3
010	0	010	0
011	2	011	3
100	-3	100	-4
101	-1	101	-1
110	2	110	2
111	3	111	3

* This is an 8-level quantization scale, with 3 representing the highest or most positive level, and -4 representing the most negative level.

C. Check of Modem Bit Synchronization and Carrier Recovery in the Presence of Heavy Intersymbol Interference

These modem functions were performed satisfactorily at the data rates of primary interest for the breadboard test. However, at rates above 350 to 370 kbits/s, some possible deterioration in these functions was observed. We believe that any such deterioration affected breadboard performance less than the 3-bit quantization limitation did. The modem indicator light "bit synch" would begin to blink at about 350 kbits/s, but observation of the modem quantized output did not indicate any major departure from performance at lower data rates. Observation of the modem quantized output was made by triggering an oscilloscope with the PN sequence epoch pulse and observing one of the output bit waveforms on the oscilloscope. Ideally, in the absence of noise, the observed waveform would be constant because of the repeating nature of the channel input. In practice, jitter occurred more often as the data rate was increased; it could have been due to bit timing problems, internal modem noise, or both, although the latter appeared more likely.

Because the modem outputs digital levels, some jitter could be expected even with excellent modem performance. This expected jitter would occur when the corresponding analog level being quantized was very close to a boundary between two digital decisions so that a very small amount of noise would make a difference.

The observed degree of jitter at high data rates and in the absence of noise seemed to indicate the occurrence of a significant modem performance deterioration. This jitter would cause the breadboard to make mistakes it would not otherwise make. However, we belive that the effects of using 3-bit quantization were more important in causing breadboard errors at high data rates.

D. Breadboard Performance Without Coding

Because of the difficulties in working with 3-bit quantization and the modem AGC, breadboard performance was not measured as in the way proposed in the test plan (Appendix F).

To obtain good performance, it is necessary that the breadboard detector have stored in it an accurate representation of the true channel response patterns. Because only 3-bit quantization was available, it was necessary for good performance that the true channel response patterns be accurately representable with 3-bit quantization. Because channel response patterns are determined from the convolution of the channel impulse response and the pulse width, the channel response patterns depend on the data rate. The constraint in allowable channel response patterns meant that only a limited number of data rates would give good performance. The set of data rates giving good performance is the set accurately representable by 3-bit uniform quantization.

The set giving good performance is further limited by the quantizing scheme built into the modem. The modem AGC continuously adjusted the level of the demodulated analog data prior to quantization. The AGC control law was such that the analog levels tended to be very high, and most of the time either the extreme positive or the extreme negative digital level was output. Thus most of the level information vital to the detector operation was lost. The situation was partly alleviated by installing a voltage divider before the quantizer. However, the modem AGC function was realized by feeding the quantization decisions back to an analog amplifier that preceded the voltage divider. Consequently, the AGC still worked to defeat the detector operation. Fortunately, the response time of the AGC control loop was on the order of 2 s. By alternately switching the voltage divider in and out of the circuit, the AGC could be overcome, and the breadboard performance measurement could be made when the levels were passing through their proper levels.

Breadboard detector operation was measured at many data rates in increments of 1 kbits/s. The performance was best at data rates of 237, 300, 340, 360, and 400 kbits/s because of the effects discussed above. The results of the measurements are shown in Tables 4 through 8. These results are summarized in Table 9, where the results obtained in the test are compared with the results predicted from the earlier study.

Table 4

BREADBOARD PERFORMANCE AT 237 kbits/s--ROM PATTERN A
(2.1-bits/cycle QPSK Equivalent)

E_b/N_o	Bit Error Rate
9.1	$6.8 \cdot 10^{-3}$
11.1	$6.8 \cdot 10^{-4}$
12.1	$1.5 \cdot 10^{-4}$
13.1	$2.5 \cdot 10^{-5}$
14.1	$3.0 \cdot 10^{-6}$

Table 5

BREADBOARD PERFORMANCE AT 300 kbits/s--ROM PATTERN B
(2.7-bits/cycle QPSK Equivalent)

E_b/N_o	Bit Error Rate
9.1	$2 \cdot 10^{-3}$
11.1	$5 \cdot 10^{-4}$
13.1	$3 \cdot 10^{-5}$
14.1	$1.1 \cdot 10^{-5}$
15.1	$1.3 \cdot 10^{-6}$

Table 6

BREADBOARD PERFORMANCE AT 340 kbits/s--ROM PATTERN A
(3.0-bits/cycle QPSK Equivalent)

E_b/N_o	Bit Error Rate
8.6	$7 \cdot 10^{-3}$
10.6	$2.5 \cdot 10^{-3}$
12.6	$4 \cdot 10^{-4}$
14.6	$7 \cdot 10^{-5}$
15.6	$2 \cdot 10^{-5}$
16.6	$3 \cdot 10^{-6}$

Table 7

BREADBOARD PERFORMANCE AT 360 kbits/s--ROM PATTERN C
(3.2-bits/cycle QPSK Equivalent)

E_b/N_o	Bit Error Rate
8.3	$2 \cdot 10^{-2}$
10.3	$5 \cdot 10^{-3}$
12.3	$2 \cdot 10^{-3}$
14.3	$3 \cdot 10^{-4}$
16.3	$1 \cdot 10^{-4}$
18.3	$7 \cdot 10^{-5}$

Table 8

BREADBOARD PERFORMANCE AT 400 kbits/s--ROM PATTERN D
(3.6-bits/cycle QPSK Equivalent)

E_b/N_o	Bit Error Rate
11.4	10^{-2}
12.4	$4 \cdot 10^{-3}$
15.4	$1.3 \cdot 10^{-3}$
17.4	$4 \cdot 10^{-4}$
19.4	$1.3 \cdot 10^{-4}$
21.4	$1.8 \cdot 10^{-5}$
23.4	10^{-5}

E. Discussion of the Results

As Table 9 shows, the results obtained at 2.7 and 3.0 bits/cycle are as predicted from the earlier study. Performance is worse than predicted for higher data rates. This is as expected, because 3-bit quantization becomes inadequate as the amount of intersymbol interference increases. A further loss may have been suffered from modem bit timing problems at these data rates.

Table 9

SUMMARY OF BREADBOARD PERFORMANCE

Data Rate (kbits/s)	QPSK Equivalent (bits/cycle)	E_b/N_o Loss at 10^{-5}-Bit Error Rate (dB)	E_b/N_o Loss at 10^{-4}-Bit Error Rate (dB)	Predicted Loss (dB)
237	2.1	4.0	3.5	1.5
300	2.7	4.0	3.5	4.0
340	3.0	5.8	5.6	5.6
360	3.2	-	8.0	
400	3.6	14.3	11.0	

The performance anomaly at 2.1 bits/cycle is also a result of quantization problems. The modem put out almost nothing but hard 1's and 0's at this data rate. The AGC suppressed the amplitude information, and the breadboard had nothing on which to base its decisions.

F. Coded Breadboard Performance

Because of interfacing problems and a malfunctioning convolutional coder, time and budget did not allow this test to be performed. However, we believe that the simulation results discussed in Section I would be obtained.

G. Conclusions

A breadboard reduced-state Viterbi detector was designed, built, and tested. Within the constraints of the modem, performance was as predicted in the earlier report.[1] The work reported here validates the predictions of the earlier study. Consequently, the improvements in performance outlined in the earlier study seem possible if a modem with the appropriate AGC were used. Testing the breadboard with an appropriate AGC would not be costly and would demonstrate bandwidth efficiencies of 2.0 to 2.5 bits/cycle at E_b/N_o of 12 to 13 dB for a bit error rate of 10^{-5}.

It was also demonstrated that combining Viterbi detection with Viterbi decoding via artificial soft decisions might be useful. Improvements over QPSK in both bandwidth efficiency and power efficiency are possible. Testing this combination would be straightforward and possibly rewarding.

Finally, because the performance of reduced-state Viterbi detection was validated, it might be useful to extend the method to other channels. The method should perform very well with other band-limited channels, such as telephone channels.

Appendix A

EXPLANATION OF THE REDUCED-STATE ALGORITHM

Appendix A

EXPLANATION OF THE REDUCED-STATE ALGORITHM

1. Description of the Reduced-State Algorithm

This appendix provides a heuristic description of the Viterbi algorithm. The description is necessary because reduced-state Viterbi detection is intimately related to the full-scale algorithm, and understanding of the former requires explanation of the latter. We regard the channel as a device that transforms discrete input pulses into waveforms that are sampled by a matched filter. Because of the noise, we can observe only a corrupted version of the sampled waveforms. This is a discrete-time problem, because the sampled outputs of a matched filter are sufficient for maximum-likelihood reception. The discussion here is nonrigorous and brief; the reader desiring a fuller discussion is directed to Ref. 1.

The channel is a finite-state device in the following sense: Because of intersymbol interference, the channel output depends on the recent n pulses, on the present pulse, and on the noise. Thus, when using binary signaling, a given pulse could produce 2^n different outputs (in the absence of noise), depending on the values of the last n input pulses. These last n pulses specify the <u>state</u> the device is in. That is, when these last n inputs are known, the input can be determined by observing the channel output: Input and state yield a unique output and a new state. Thus, the problem of estimating the input pulses is identical to the problem of tracking the channel through its states.

In the absence of noise, observation of the outputs of the channel would allow determination of the sequence of states through which the channel has proceeded and hence would enable determination of the information input. Unfortunately, because noise in the channel allows observation of only the corrupted outputs, one cannot calculate with certainty the sequence of states. However, because the channel has only a finite number of possible states, one can calculate the probability that it has followed each of the possible sequences of states when one is given the noisy output. The maximum-likelihood estimate of the input information sequence then corresponds to the input sequence that drives the channel through the series of states found to have the greatest probability. Thus, in general, maximum-likelihood estimation requires examination of the noiseless output corresponding to each possible input, and subsequent calculation of the likelihood that the difference between the noiseless output and the actually observed output could have been caused by noise. This is a complicated process, because the number of possible state sequences grows exponentially with time. Fortunately, the finite-state structure of the channel greatly simplifies the calculation by eliminating output sequences from consideration as channel output data are received. The basic concepts of this elimination are:

- The output of the channel depends only on the state of the channel and its most recent input.
- Therefore, if the present state of the channel is known with certainty, future output can tell us nothing about the sequence of states the channel has followed to the present state.
- Hence, in calculating the required likelihoods, we need consider only the most likely sequence of states to each of the possible present states.

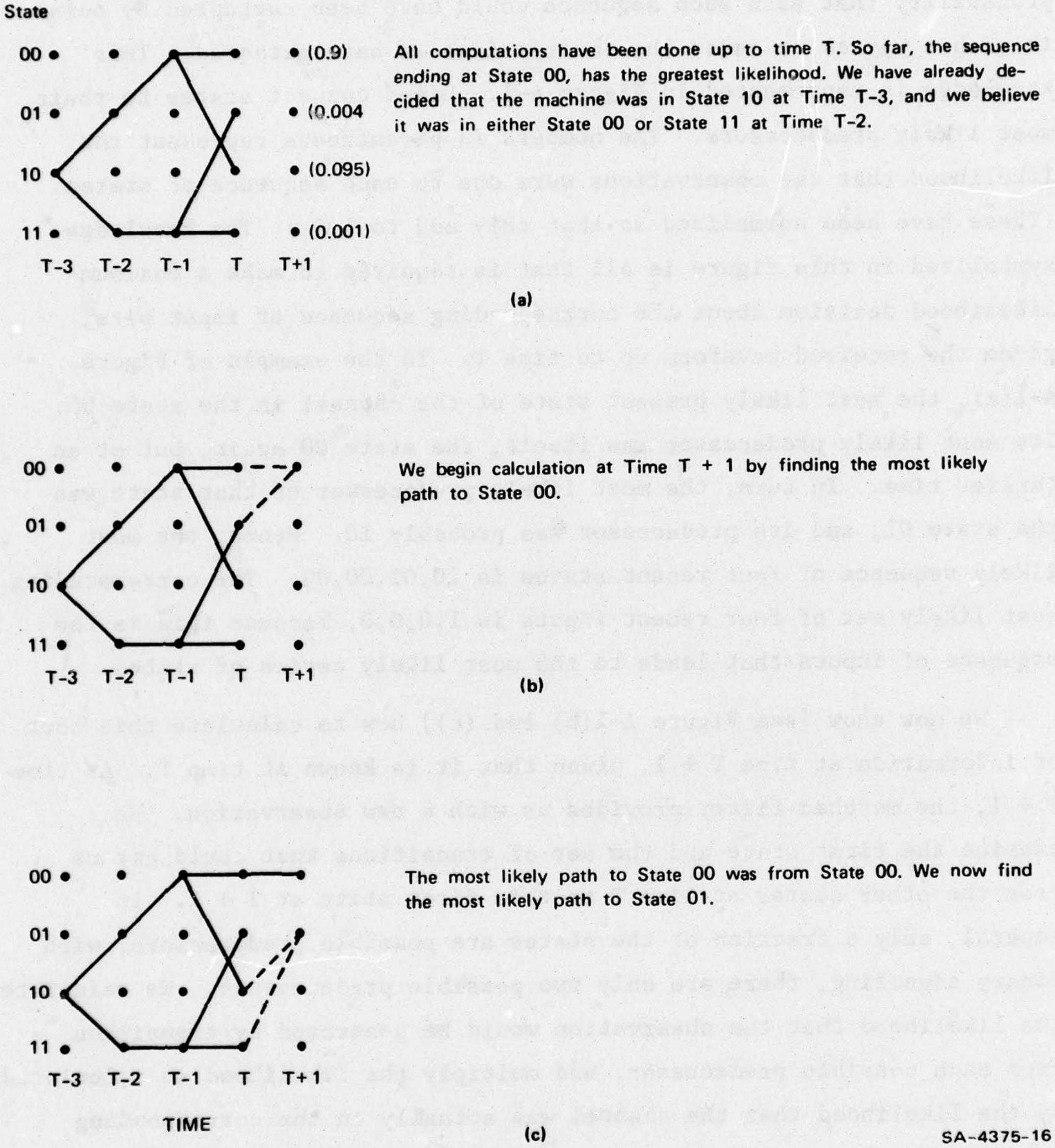
Stated differently, because all other sequences may be discarded, and because future data can do nothing to change our estimates of these most likely sequences, the only uncertainty about history that can still be resolved is represented by uncertainty about the actual state the channel is in at present.

The Viterbi algorithm is based on the above concepts. The key idea is that, at any time T, we can eliminate some sequences of states from further consideration, because we know they cannot be maximum-likelihood sequences. This can be done, even though all the data from the channel have not yet been received. We now detail the computations.

2. Examples

Suppose that at time T we know the most likely sequence of states leading to each of the possible states. Suppose that we also know the probability that each such sequence could have been corrupted by noise in such a way as to cause the observations we have gathered. This knowledge is represented in Figure A-1. Lines connect states to their most likely predecessors. The numbers in parentheses represent the likelihood that the observations were due to each sequence of states. (These have been normalized so that they add to 1.0.) The knowledge symbolized in this figure is all that is required to make a maximum-likelihood decision about the corresponding sequence of input bits, given the received waveform up to time T. In the example of Figure A-1(a), the most likely present state of the channel is the state 00. Its most likely predecessor was itself, the state 00 again, but at an earlier time. In turn, the most likely predecessor of that state was the state 01, and its predecessor was probably 10. Hence, the most likely sequence of four recent states is 10,01,00,00. The corresponding most likely set of four recent inputs is 1,0,0,0, because this is the sequence of inputs that leads to the most likely series of states.

We now show [see Figure A-1(b) and (c)] how to calculate this sort of information at time T + 1, given that it is known at time T. At time T + 1, the matched filter provides us with a new observation. We examine the first state and the set of transitions that could get us from the other states at time T to this first state at T + 1. In general, only a fraction of the states are possible predecessors; with binary signaling, there are only two possible predecessors. We calculate the likelihood that the observation would be generated by transition from each possible predecessor, and multiply the likelihood s calculated by the likelihood that the channel was actually in the corresponding predecessor state. In Figure A-1, this yields the likelihood that the observations originate from each of the hypothesized trajectories illustrated. We select the single trajectory having the highest likelihood and store it, together with the calculated likelihood; the others we can forget. Since the two hypothesized trajectories lead to the



All computations have been done up to time T. So far, the sequence ending at State 00, has the greatest likelihood. We have already decided that the machine was in State 10 at Time T-3, and we believe it was in either State 00 or State 11 at Time T-2.

We begin calculation at Time T + 1 by finding the most likely path to State 00.

The most likely path to State 00 was from State 00. We now find the most likely path to State 01.

FIGURE A-1 VITERBI ALGORITHM

same state, future observations will not provide further reason for choosing between the two trajectories.

When we have finished the calculations for the first state, we proceed with those for the second, and so on. Theoretically, the final maximum-likelihood sequence could be determined only after the entire message sequence had been transmitted. Actually, the maximum-likelihood paths to each state as calculated above tend to merge; the maximum-likelihood path up to any point in the received sequence can be determined with relatively little delay. Such merging is illustrated in Figure A-1.

3. Summary

The preceding has described the basic Viterbi algorithm. A number of comments about it are in order:

- To reiterate, the algorithm performs as well as any possible algorithm in most detection problems.
- Calculation of the probabilities need not entail multiplication. One can use the log of probability throughout and add. This does not affect the validity of the comparison operation, because log is a monotonic function.
- Except for the comparison and selection of competing trajectories after the probabilities have been calculated, the entire algorithm can be done in parallel.
- Errors propagate for only a few tens of bits at most.

The reduced-state version of the algorithm is now simple to explain. If the reader will examine Figure A-1 once more, he will notice that the state 00 has likelihood 0.9, and the state 10 has likelihood 0.095. Thus, the likelihood that the channel is in either one of these two states is 0.995. In other words, the channel is almost certainly in one of these two states. This illustrates a rule that is generally true:

> At any time period, one can examine the calculated likelihoods and specify a miniscule set of states in which the channel will be, with very high probability.

(Indeed, the likelihoods shown in Figure A-1 are unduly pessimistic. It would be more representative of normal operation if the sum of the likelihoods of the two most likely states were 0.99999, rather than 0.995.) As a consequence of the above rule, performance will be only slightly degraded if the algorithm assumes that the channel is certainly in one of these most likely states, and if it ignores other possible predecessor states in its calculations. This is the reduced-state algorithm. The device incorporating the algorithm remembers which states the channel is probably in, and does its calculations only for these states and their successor states. (This list of most likely states constantly changes, of course.) For the example of Figure A-1, where the number of states the full Viterbi algorithm uses is four, not much is gained by adopting the reduced-state algorithm: The number of required calculations is cut only in half. In real band-limited channels, however, significant intersymbol interference may extend over many signaling periods, and the saving in calculations required is much more impressive.

Appendix B

ANALYSIS OF BAND-LIMITED PERFORMANCE

Preceding Page BLank - NOT FILMed

Appendix B

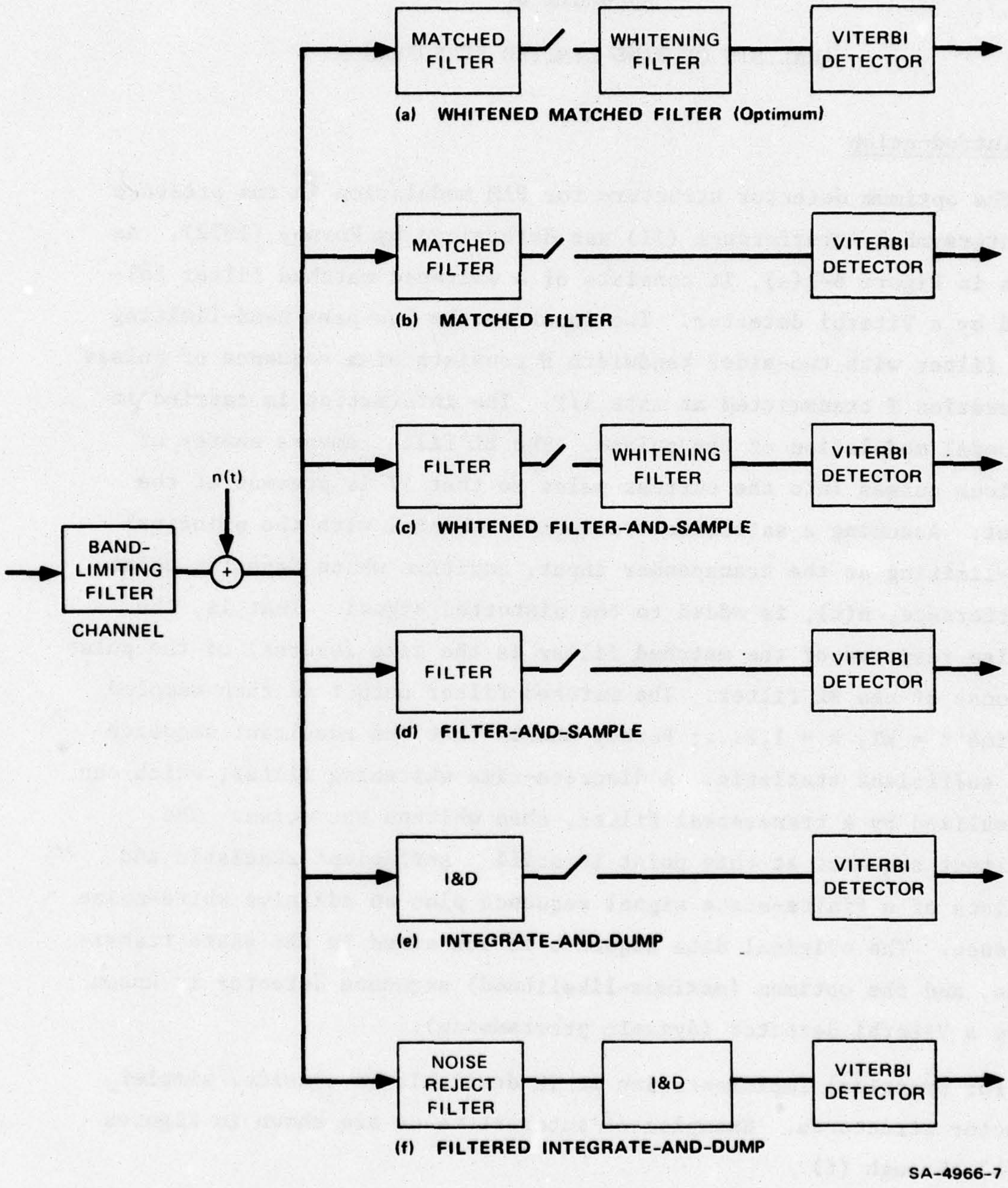
ANALYSIS OF BAND-LIMITED PERFORMANCE

1. Introduction

The optimum detector structure for PAM modulation in the presence of intersymbol interference (II) was determined by Forney (1972). As shown in Figure B-1(a), it consists of a whitened matched filter followed by a Viterbi detector. The input to the low-pass band-limiting (BL) filter with two-sided bandwidth B consists of a sequence of pulses of duration T transmitted at rate 1/T. The information is carried in antipodal modulation of the pulses. The BL filter smears energy of previous pulses into the current pulse so that II is present at the output. Assuming a satellite transponder channel with the principal band-limiting at the transponder input, additive white Gaussian noise interference, n(t), is added to the distorted signal. That is, the impulse response of the matched filter is the time reversal of the pulse response of the BL filter. The matched filter output is then sampled at time $t = kT$, $k = 1,2...$; Forney showed that the resultant sequence is a sufficient statistic. A discrete-time whitening filter, which can be realized by a transversal filter, then whitens the noise. The resultant sequence at this point is still a sufficient statistic and consists of a finite-state signal sequence plus an additive white-noise sequence. The original data sequence is contained in the state transitions, and the optimum (maximum-likelihood) sequence detector is known to be a Viterbi detector (dynamic programming).

For practical implementation it is desirable to consider simpler detector structures. Examples of interest to us are shown in Figures B-1(b) through (f).

In Figure B-1(b), the detector is simply the optimum detector with the whitening filter removed. The Viterbi algorithm is a maximum-likelihood sequence detector only when the noise sequence is white, and



(a) WHITENED MATCHED FILTER (Optimum)

(b) MATCHED FILTER

(c) WHITENED FILTER-AND-SAMPLE

(d) FILTER-AND-SAMPLE

(e) INTEGRATE-AND-DUMP

(f) FILTERED INTEGRATE-AND-DUMP

FIGURE B-1 SEQUENCE DETECTORS

it suffers some degradation when the noise is colored. We use this detector to describe the effects of colored noise on Viterbi detector performance.

A potential difficulty associated with implementation of the optimum-sequence detector is synthesis of the matched filter. Letting $S(f) = |S(f)| \exp \{j\theta(f)\}$ denote the transform of the pulse response of the BL filter, the matched-filter transform is given by $G(f) = |S(f)| \exp \{-j\theta(f)\}$. Although the amplitude response S(f) can be approximated, simultaneous synthesis of the phase response, $-\theta(f)$, may be difficult with the RF components required for high data rate operation. Since the degradation caused by a detector filter with a matched amplitude response but a mismatched phase response is not known, the filter-and-sample (F&S) detector of Figure B-1(d) is of interest. There the detector filter is assumed to be the same as the BL filter. For high data rates ($1/BT >> 1$), the amplitude response of the detector filter will then be approximately matched and the phase response be mismatched.

We are most interested in the integrate-and-dump (I&D) detector of Figure B-1(e). At high rates the I&D detector passes approximately 1/BT times as much noise as the F&S detector. On the other hand, the output noise samples of the I&D detector are white, whereas those of the F&S detector are correlated. Recall that correlated noise implies some Viterbi detector degradation. We might consider insertion of a noise-rejection filter in front of the integrator of the I&D detector. Specification of the bandwidth of this filter might involve trade-off of decreased detector noise for increased noise correlation, and performance would lie somewhere between that for the I&D detector and that for the F&S detector.

We now examine performance of selected detectors from Figure B-1. So that the results will apply to a satellite transponder channel, we describe performance in terms of an effective signal-to-noise ratio (SNR) E/N_o, where E corresponds to the energy in the distorted pulse at the BL filter output, and N_o is the one-sided noise spectral density. The theory developed applies to any antipodal PAM modulation scheme.

This includes MSK as well as QPSK or BPSK. For MSK the pulses are sinusoidal, whereas for QPSK or BPSK the pulses are rectangular.

2. Optimum Detector

Next, we briefly review parts of Forney's work on the optimum detector, for later reference. Let H(t) be the pulse response of the BL filter. Then the distorted signal at the BL filter output is the superposition of filter pulse responses given by

$$s(t) = \sum_{k=-\infty}^{\infty} u_k \, H(t - kT) \quad , \qquad \text{(B-1)}$$

where $u_k = \pm 1$ is the data sequence. For convenience, we normalize H(t), so that energy is unity in a distorted pulse:

$$\int_{-\infty}^{\infty} H^2(t)\, dt = 1 \ .$$

Correspondingly, we must scale the noise, n(t), to have spectral density, ρ, given by

$$\rho = (E/N_o)^{-1} \ .$$

Letting h(t) be the impulse response of the matched filter, the matched-filter output at the sampling instants is given by

$$x_k = \int_{-\infty}^{\infty} h(kT - \tau) \; [s(\tau) + n(\tau)] \; d\tau \quad .$$

Substituting Eq. (B-1), and noting that, for the matched filter, h(t) = H(-t),

$$x_k = \sum_{j=-\infty}^{\infty} r_{k-j}\, u_j + \nu_k$$

$$r_k = \int_{-\infty}^{\infty} H(t)\, H(t + kT)\, dt$$

$$\nu_k = \int_{-\infty}^{\infty} H(t - kT)\, n(t)\, dt \quad . \qquad \text{(B-2)}$$

The noise sequence $\{\nu_k\}$ is correlated with covariance

$$\overline{\nu_k \nu_j} = \int_{-\infty}^{\infty}\int_{\infty}^{\infty} H(t - kT)\, H(\tau - kT)\, \overline{n(t)\, n(\tau)}\, dt\, d\tau$$

$$= \frac{\rho}{2}\, r_{k-j} \quad . \qquad \text{(B-3)}$$

We truncate the pulse response H(t) so that it vanishes for $t \geq (L+1)\, T$, where L is sufficiently large that the truncation energy loss is negligible. From Eq. (B-3), this implies that the noise has finite memory

$$r_k = 0 \qquad \text{for} \qquad |k| \geq L \quad .$$

Then Eq. (B-2) can be written in delay transform notation:

$$x(D) = u(D)\, r(D) + \nu(D)$$

$$u(D) = \sum_{k=-\infty}^{\infty} u_k D^k$$

$$r(D) = \sum_{k=-L}^{L} r_k D^k$$

$$\nu(D) = \sum_{k=-\infty}^{\infty} \nu_k D^k \quad , \qquad \text{(B-4)}$$

where $\nu(D)$ is a colored-noise sequence with autocorrelation function $(\rho/2)\, r(D)$. Forney's key result was proof that $x(D)$, the output sample sequence of the matched filter, is a sufficient statistic. In principle, a maximum-likelihood sequence detector can be applied directly to observation $x(D)$, but in practice this is computationally prohibitive. If $x(D)$ is passed through a whitening filter that decorrelates $\nu(D)$, the output is still a sufficient statistic. Moreover, the resultant white-noise sequence allows the use of the efficient Viterbi algorithm to compute the ML sequence.

First, we note that the coefficients of $(\rho/2)\, r(D)$ correspond to the autocorrelation function of the noise sequence $\nu(D)$. It is well known that $r(D)$ can be decomposed as

$$r(D) = f(D) f(D^{-1}) \quad , \qquad \text{(B-5)}$$

where $1/f(D^{-1})$ is a whitening filter for $\nu(D)$. That is, if

$$n(D) = \nu(D)/f(D^{-1}), \qquad \text{(B-6)}$$

then n(D) is a white-noise sequence with variance ρ/2. Now, applying the whitening operation to Eq. (B-4),

$$\begin{aligned} y(D) &= x(D)/f(D^{-1}) \\ &= u(D)f(D) + n(D) \quad , \qquad \text{(B-7)} \end{aligned}$$

which is Forney's discrete-time representation. Since n(D) is a white-noise sequence, and f(D) is a finite-state machine driven by the data sequence u(D), the Viterbi algorithm is applicable. The whitening operation is nonsingular, so y(D) is also a sufficient statistic.

The minimum (Euclidean) distance of the distorted signal is defined as

$$\begin{aligned} d_{min} &= \underset{\underline{u} \neq \underline{u}'}{\text{Min}} \; ||u(D)\; f(D) - u'(D)\; f(D)|| \\ &= \underset{\underline{u} \neq \underline{u}'}{\text{Min}} \; ||[u(D) - u'(D)]\; f(D)|| \\ &= \underset{\underline{v}\, \varepsilon\, a}{\text{Min}} \; ||v(D)\; f(D)|| \quad , \qquad \text{(B-8)} \end{aligned}$$

where a is the set of all finite-order polynominals with coefficients taken from the alphabet (+2,-2,0), and having a nonzero constant coefficient.

Forney showed that for high SNR the error-event probability is approximately

$$P_e \cong K \cdot Q\left(\frac{d_{min}}{2} / \sqrt{\rho/z}\right) \quad , \qquad \text{(B-9)}$$

where K is the number of signals at the minimum distance from the correct signal and

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t2/2}\, dt$$

A physical explanation for this is as follows. At high SNR an error event will almost always involve a detector decision in favor of an incorrect sequence at minimum distance from the correct sequence. Since the noise is white, its distribution is spherically symmetric in all directions in Euclidean space. In particular, it has variance ρ/z along any line joining the correct sequence and an incorrect sequence. Then the probability of deciding in favor of any incorrect sequence at minimum distance from the correct sequence is the probability that the noise along the common line exceeds half the minimum distance (ML detection is minimum-distance detection). That is, the incorrect pairwise decision in favor of the incorrect sequence is $Q\left(\frac{d_{min}}{z}/\sqrt{\rho/z}\right)$. Since there are K such incorrect sequences at the minimum distance from a correct sequence, a union upperbound gives Eq. (B-9).

Both d_{min} and K are determined by computation of (B-8) by exhaustive search. To summarize:

- Compute the pulse response of the BL filter, normalize to unit energy, and denote H(t).
- Truncate H(t) to $0 \leq t \leq (L + 1)T$, with L large enough that the energy lost is negligible.
- Compute the coefficients of r(D) given by Eq. (B-2), which can be determined by sampling the output of the matched filter at time kT, k = 1,2,... for input H(t).
- Compute the coefficients of f(D) in terms of the coefficients of r(D), as described in Appendix C.
- Compute d_{min} according to Eq. (B-8), and find the performance by Eq. (B-9).

3. Viterbi Detection in Colored Noise

The Viterbi algorithm is the ML detector only when the additive noise is white. Suppose the whitening filter is removed from the optimum detector, as shown in Figure B-1(b). From Eq. (B-2), we have the observation

$$x(D) = u(D)\, r(D) + \nu(D) \quad ,$$

where $\nu(D)$ is colored. The minimum signal distance in Euclidean space is then given by Eq. (B-8) with $f(D)$ replaced by $r(D)$:

$$d_{min} = \operatorname*{Min}_{\underline{v}\,\epsilon\, a} \|\nu(D)\; r(D)\| \quad . \qquad \text{(B-10)}$$

Since the noise sequence, $\nu(D)$, is colored with autocorrelation function $\frac{\rho}{z}\, r(D)$, its distribution is no longer spherically symmetric, and its variance along a line joining the correct sequence and a minimum-distance sequence depends on the direction of that line. An upper bound using the largest variance can be used for a worst-case analysis. The largest variance occurs in the direction of the eigenvector corresponding to the largest eigenvalue of the covariance matrix for $\nu(D)$. Let λ_{max} be the largest eigenvalue of the matrix R, where

$$R = [r_{ij}]$$

$$r_{ij} = r_{i-j} \quad . \qquad \text{[Defined by Eq. (B-2).]}$$

The noise covariance matrix is $(\rho/z)R$. Then the error event probability is approximately

$$P_e \cong K\, Q\left(\frac{d_{min}}{z} \Big/ \sqrt{\frac{\rho\lambda_{max}}{z}}\right) \quad ,$$

where d_{min} is given by Eq. (B-10).

4. Filter-and-Sample Detector

It is of interest to examine detector filters having amplitude responses approximately matched to the signal but having mismatched phase responses. Derivation of performance is the same as for the matched filters, except that the detector filter impulse response $h(t)$ and the BL filter pulse response $H(t)$ are no longer related. A sufficient statistic is given by

$$x_k = \sum_{j=-\infty}^{\infty} a_{k-j}\, u_j + \nu_k \quad ,$$

where

$$a_k = \int_{-\infty}^{\infty} h(t)\, H(kT - t)\, dt$$

$$\nu_k = \int_{-\infty}^{\infty} h(t - kT)\, n(t)\, dt \quad ,$$

or

$$x(D) = a(D)\, u(D) + \nu(D) \quad .$$

The noise sequence $\nu(D)$ is colored with covariance

$$\overline{\nu_k\, \nu_j} = \frac{\rho}{z}\, r_{k-j} \quad ,$$

where

$$r_k = \int_{-\infty}^{\infty} h(t)\, h(t - kT)\, dt \quad .$$

We assume the detector filter gain is normalized to give

$$\int_{-\infty}^{\infty} h^2(t)\, dt = 1 \quad .$$

This implies that $\nu(D)$ has variance ρ/z. Although the noise sequence is colored, we can, as in the case of the matched filter detector, apply the Viterbi algorithm and suffer the loss due to the noise correlation. Then the error event probability is approximated at high SNR by

$$P_e \cong K\,Q\left(\frac{d_{min}}{z} \Big/ \sqrt{\frac{\rho\lambda_{max}}{z}}\right) ,$$

where

$$d_{min} = \min_{\underline{v}\varepsilon\, a} ||v(D)\; a(D)|| ,$$

and λ_{max} is the largest eigenvalue associated with a Toepliz matrix with entries of the k^{th} diagonal given by r_k. This gives the performance of the detector in Figure B-1(d).

As in the matched filter case, we can whiten the observation. The resultant is still a sufficient statistic, and the Viterbi algorithm is then a maximum-likelihood sequence detector. That is, decompose

$$r(D) = f(D)\; f(D^{-1}) .$$

Then, $1/f(D^{-1})$ is a whitening filter for $\nu(D)$. A new sufficient statistic, y(D), is formed by

$$\begin{aligned} y(D) &= x(D)/f(D^{-1}) \\ &= u(D)\; a(D)/f(D^{-1}) + n(D) \\ &= u(D)\; g(D) + n(D) , \end{aligned}$$

where n(D) is white. The error event probability is approximately

$$P_e \cong K\,Q\left(\frac{d_{min}}{z} \Big/ \sqrt{\rho/z}\right) ,$$

where

$$d_{\min} = \underset{\underline{v}\epsilon\mathcal{A}}{\mathrm{Min}} ||v(D)\ g(D)|| \ .$$

This gives the performance for the detector of Figure B-1(c).

5. Integrate-and-Dump Detector

We now consider the I&D detector of Figure B-1(e). The BL filter output is the distorted signal

$$s(t) = \sum_{k=-\infty}^{\infty} u_k\ H(t - kT) \quad ,$$

where H(t) is the pulse response of the BL filter. The distorted signal plus noise is integrated over a pulse duration, sampled, and then dumped. The resultant sequence is

$$y_k = \sum_{j=0}^{\infty} u_{k-j}\ b_j + n_k \quad ,$$

where

$$b_j = \frac{1}{\sqrt{T}} \int_o^T H(t + jT)\ dt$$

$$n_k = \frac{1}{\sqrt{T}} \int_o^T n(t + kT)\ dt \quad .$$

The noise sequence $\{n_k\}$ is independent, identically distributed with variance

$$\overline{n_k^2} = \frac{1}{T} \int_o^T \int_o^T \overline{n(t + kT)\ n(\tau + kT)}\ dt\ d\tau = \rho_{\overline{z}} \quad .$$

The error event probability for the I&D sequence detector of Figure B-1(e) is then given by the approximation

$$P_e \cong K\, Q\left(\frac{d_{min}}{z} / \sqrt{\rho/z}\right) ,$$

where

$$d_{min} = \min_{\underline{v} \epsilon a} ||v(D)\ b(D)|| \quad .$$

For low signaling rate (1/BT << 1), the I&D detector is asymptotically optimum.

6. Comparison of Detector Performance

For each of the detectors we have carefully defined d_{min} in such a way that the error event probability is given by the high-SNR approximation

$$P_e \cong K\, Q\left(\frac{d_{min}}{z} / \sqrt{\frac{\rho\lambda_{max}}{z}}\right)$$

for all the detectors, where

$\rho = (E/N_o)^{-1}$

λ_{max} = the largest eigenvalue of the detector output noise covariance matrix

K = the number of minimum-distance sequences.

For comparison of detector performance, define

$$d^*_{min} = d_{min}/\sqrt{\lambda_{max}} \quad .$$

The expression for P_e then becomes

$$P_e \cong K\,Q\left(\frac{d^*_{min}}{z}\cdot\sqrt{\frac{zE_b}{N_o}}\right) .$$

Since P_e depends only on d^*_{min} and the SNR, we can compare detector performance in terms of d^*_{min}.

We compare here the performance of the following sequence detectors: matched filter, F&S, and I&D. The performance of the other detectors in Figure B-1 can be estimated from the results of these comparisons. The BL filter is a 9-pole Chebyschev filter. An identical filter is used in the F&S detector. In Figure B-2 we compare d_{min} as a function of normalized rate. Band-limiting bandwidth B is one-sided, and the Nyquist rate corresponds to 1/BT = 2. At high rates the matched filter performance and the F&S detector performance are nearly the same. The reason for this is that at high data rates the matched filter impulse response is approximately the time reversal of the F&S detector filter impulse response. The mainlobe of the impulse response of a 9-pole Chebyschev filter is nearly symmetrical, so the effect of this time-reversal relation is negligible. For the case of an ideal BL filter having a perfectly symmetrical sinc function impulse response, the theoretical performance of the matched filter and that of the F&S detector are asymptotically equivalent at high data rates. At low rates, d_{min} decreases for the F&S detector because of our assumption that the detector bandwidth is equal to the BL filter bandwidth. At low rates, best performance is achieved by setting the bandwidth of the F&S detector approximately equal to that of the BL filter input signal (i.e., one-sided detector bandwidth approximately equal to 1/2T).

For both the matched filter and F&S detectors, the noise samples are colored. In Section 3 we have performed a worst-case analysis of correlated noise effects on Viterbi algorithm performance which has shown that performance loss in terms of signal energy is proportional to

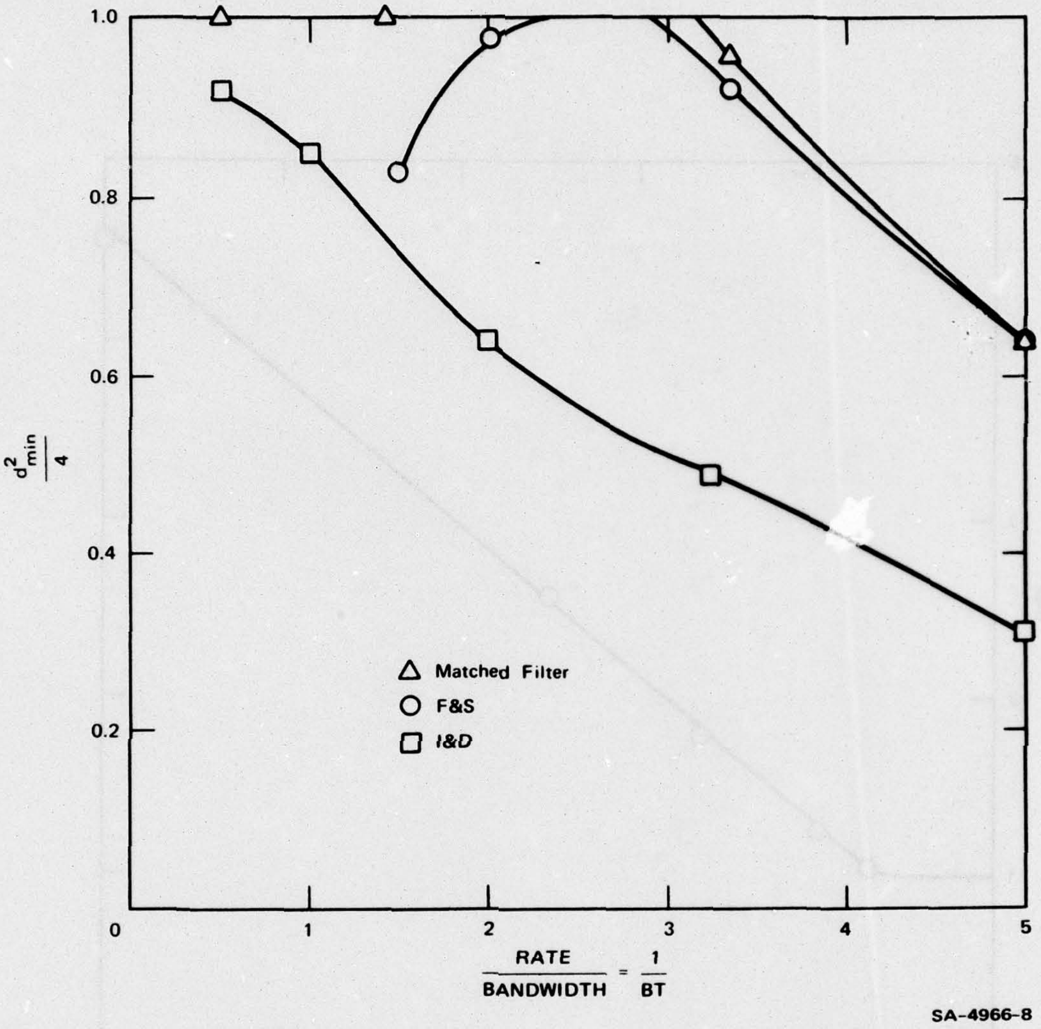
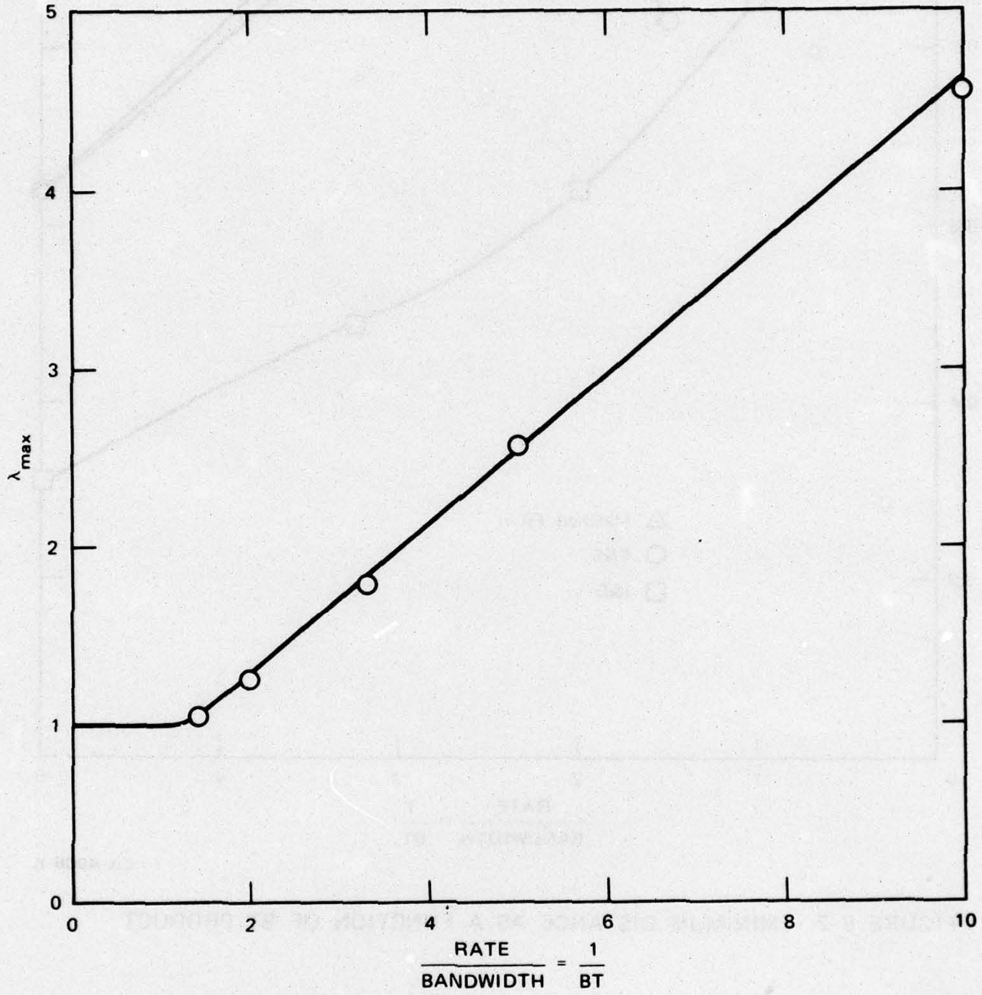


FIGURE B-2 MINIMUM DISTANCE AS A FUNCTION OF BT PRODUCT

the largest eigenvalue λ_{max} of the colored-noise covariance matrix. The behavior of λ_{max} is shown in Figure B-3. The minimum distance adjusted to include the noise correlation effects is shown in Figure B-4. In Figure B-5 the results are expressed in terms of the increased signal power required to maintain a constant error rate.



SA-4966-9

FIGURE B-3 LARGEST EIGENVALUE OF COLORED-NOISE COVARIANCE MATRIX

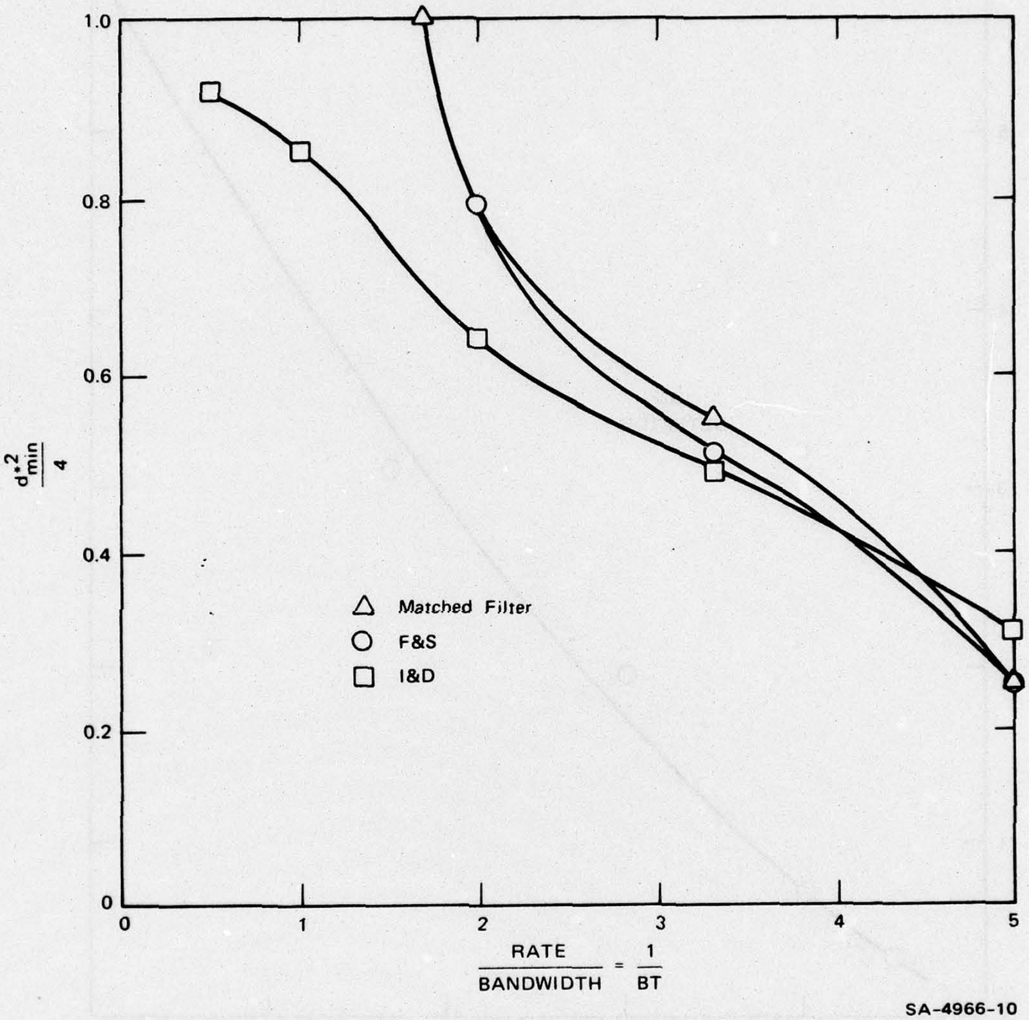
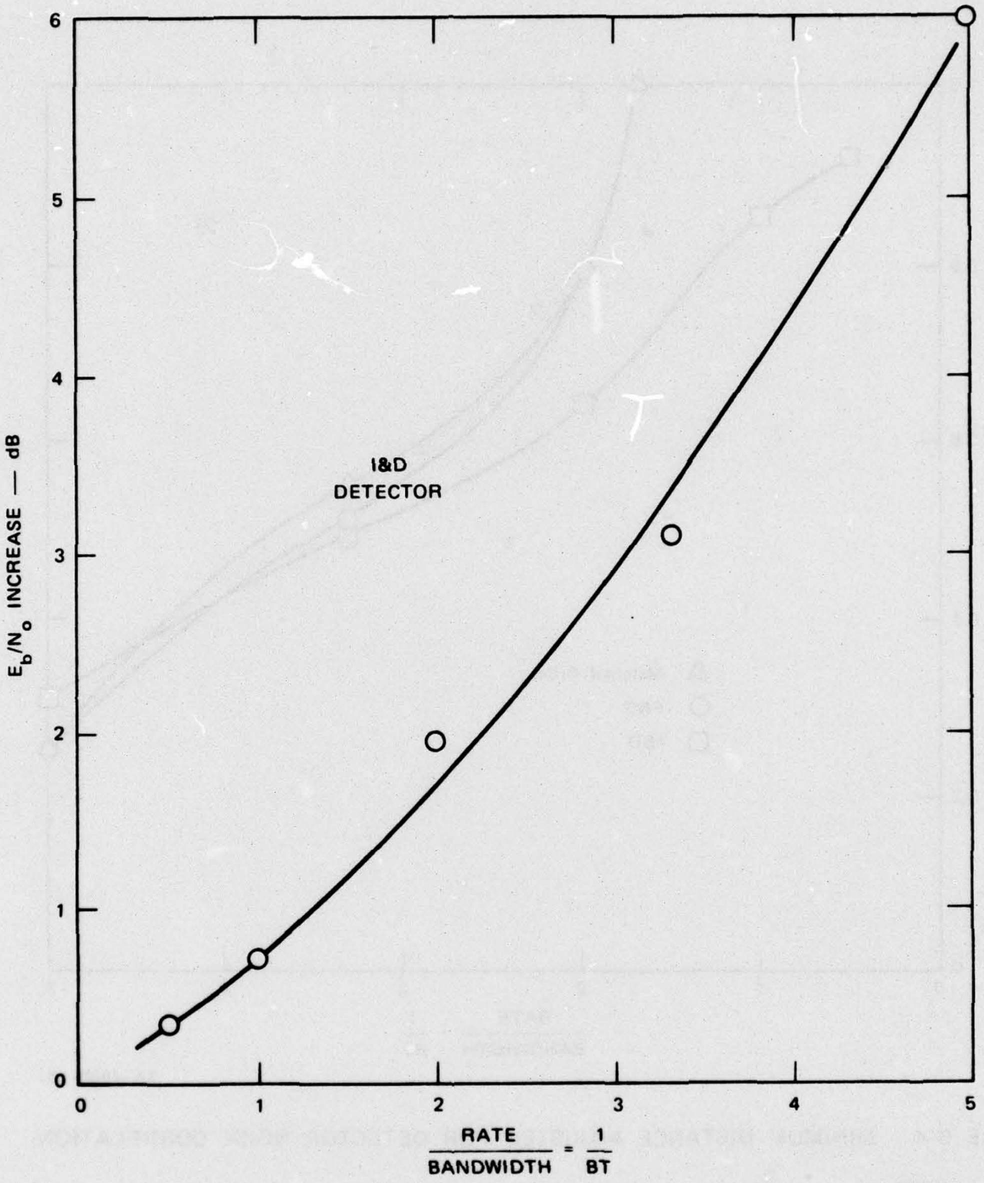


FIGURE B-4 MINIMUM DISTANCE ADJUSTED FOR DETECTOR NOISE CORRELATION



SA-4966-11

FIGURE B-5 INCREASED E_b/N_o REQUIRED—INTEGRATE-AND-DUMP

Appendix C

COMPUTATION OF f(D)

Preceding Page Blank - Not Filmed

Appendix C

COMPUTATION OF f(D)

In Appendix B [Eq. (B-5)] the optimum detector minimum distance is given in terms of f(D), where

$$r(D) = f(D)f(D^{-1}) \quad ,$$

and $\rho_z r(D)$ is the noise autocorrelation function of the matched filter output. Here r(D) has the form

$$r(D) = \sum_{k=-L}^{L} r_k \, D^k$$

and is symmetric; that is, $r_k = r_{-k}$. Then r(D) has the factorization

$$r(D) = C \prod_{i=1}^{L} (a_i + D) \left(a_i + D^{-1}\right) \quad .$$

where c is a constant, and a_i and a_i^{-1} are roots of r(D). Thus we can define

$$f(D) = \sqrt{C} \prod_{i=1}^{L} (a_i + D) \quad . \tag{C-1}$$

To find the roots of r(D), let

$$s(D) = D^k \, r(D)$$

$$= \sum_{k=0}^{2K} S_k \, D^k \quad .$$

Then s(D) is a polynomial of order 2K,

$$s_k = r_{k-K} \quad ,$$

and

$$s(D) = \left(C \sum_{i=1}^{K} a_i \right) \overset{K}{\underset{i=1}{\pi}} \; (D + a_i) \left(D + \frac{1}{a_i} \right) \quad . \qquad \text{(C-2)}$$

For the optimum detector we then use the following method for finding f(D):

(1) Convolve the unit-energy-distorted pulse response of the BL filter with its time reversal, and sample at times t = kT. This gives r(D).

(2) Compute the roots of s(D) by standard root-finding algorithms. This gives $\{a_i\}$ and C, from Eq. (C-2).

(3) Compute f(D) from Eq. (C-1).

Appendix D

WHITENING FILTERS

Appendix D

WHITENING FILTERS

Some of the detectors employ whitening filters. We give one form (not unique) of a whitening filter for an arbitrary colored-noise sequence with autocorrelation function r(D).

A colored-noise sequence can be whitened (orthonormalized in the appropriate Hilbert space) by a Gram-Schmidt orthonormalization procedure for any finite sequence. The whitening filter in this case is a time-varying transversal filter. By taking infinitely long messages, or equivalently assuming steady-state whitening-filter operation, the whitening filter can be shown to be a time-invariant transversal filter.

In the Gram-Schmidt procedure, by taking noise vectors backward/forward in time, one obtains a purely noncausal/causal whitening filter. Since the procedure is straightforward, we simply give the form of the causal whitening filter. For a colored-noise input sequence V(D) with autocorrelation function r(D), the causal whitening filter is given by the transversal filter shown in Figure D-1.

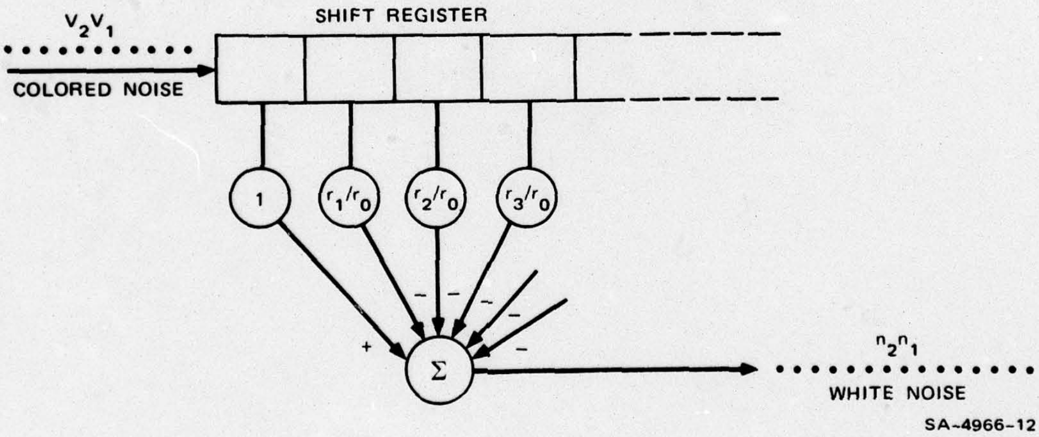


FIGURE D-1 TRANSVERSAL FILTER

The i^{th} tap connection gain is given by r_{i-1}/r_o . For any practical system, r(D) can be truncated, so a finite-length filter can be used.

Preceding Page BLANK - NOT FILMED

Appendix E

REDUCED-STATE VITERBI DETECTOR TEST PLAN

Preceding Page Blank - NOT FILMed

Appendix E

REDUCED-STATE VITERBI DETECTOR TEST PLAN

1. Objectives

The objectives of this test plan are as follows:

(1) To verify that the reduced-state Viterbi detector breadboard performance conforms with the predicted performance.

(2) To demonstrate that use of the reduced-state Viterbi detector can lead to significant gains in band-limited channel throughput.

(3) To determine whether reduced-state Viterbi detectors can be retrofitted to existing DCA equipment, such as the MD921/G modem, to improve band-limited satellite throughput.

2. Main Tests

The above objectives will be realized through the accomplishment of four basic tests. The first test will measure the normal band-limited performance of the MD921/G biphase modem without the reduced-state Viterbi detector breadboard. This performance will establish a baseline with which the improvement given by the breadboard can be compared. The measurements will be made with the 225-kHz band-limiting filter and with data rates up to the Nyquist data rate. At each data rate, performance will be measured by finding the bit error rates as a function of the energy per bit divided by the noise density ratio (E_b/N_o).

The second test we will perform is one that will look for possible implementation problems of the MD921/G modem that would adversely effect breadboard performance. It may be that the modem will not perform adequately at very high bit rates, because it was not designed to work in the presence of large amounts of intersymbol interference. Carrier-recovery and bit-timing problems may be found. It would be very important to be aware of such problems before the breadboard test is conducted, to avoid mistaking modem performance problems for breadboard performance problems.

The third test will examine the performance of the modem-breadboard combination. This examination of performance will be done at several values of E_b/N_o and at several values of data rate. No error-correction coding will be used. This test will be the primary test showing the value of the reduced-state Viterbi detector concept for improvement of hand-limited throughput.

The fourth test will be similar to the third but will measure the performance of the breadboard detector when Viterbi decoding and rate one-half convolutional codes are used. The performance of the combination of the breadboard, the modem, and the KY801 coder will be measured both when the breadboard puts out hard decisions and when it puts out 3-bit artificial soft decisions. In this test we will also examine the conditional distributions of the 3-bit artifical soft decisions, given the input data bit. We will use these conditional distributions to estimate the performance of the breadboarding in conjunction with the coder if a randomizing interleaving were used.

3. Parameters To Be Varied

In conducting the above tests the following parameters of the test setup will be varied:

(1) The energy per bit will be changed by varying the signal power from the modem through a step attenuator.

(2) The data rate output by the modem will be varied by changing the switches on the front of the modem.

(3) The channel-response read-only memories (ROMs) in the breadboard will be changed to match the required data rate. The channel-response ROMs are required for each data rate used.

(4) Finally, various equipment connections will be changed for each of the above tests.

4. Parameters To Be Measured

In conducting the above tests several parameters must be measured.

a. Measurement of E_b/N_o

The most important of the parameters to be measured is E_b/N_o. The test setup shown in Figure E-1 will be used. E_b/N_o will be measured at the output of the 225-kHz band-limiting filter, because this is where the signal power corresponding to satellite output is located. Energy per bit and noise density will be measured separately. N_o will be measured by using a standard noise-power-measuring instrument at the receiver (demodulator) input. This noise-power instrument uses a filter with known equivalent noise bandwidth and measures the noise power that passes through the filter. E_b will be calculated from a measurement of signal power filter output. The signal power will be measured by connecting the filter to an rms power meter. E_b will then be calculated from this figure by dividing the recorded rms power by the data rate in bits per second. An E_b measurement will be made for each data rate used, since the output waveform may change as the data rate changes. While making these measurements of E_b/N_o we will construct a table of instrument settings that will enable us to configure the test setup for various values of E_b/N_o at a later date. We will not need to remeasure E_b/N_o for every test; rather, we will consult the table and return instruments to the settings required to reach a given E_b/N_o. However, for each run of data we will recalibrate a few of the E_b/N_o values, to ensure accuracy.

b. Measurement of Data Rate and Channel Response

The data rate will be taken from the thumbwheel switches on the front of the modem.

The breadboard detector requires that the channel response for any given data rate be known. We will measure this channel response and use it to program to the ROMs that the breadboard detector will use in its operation. Measurement of channel response is complicated by the

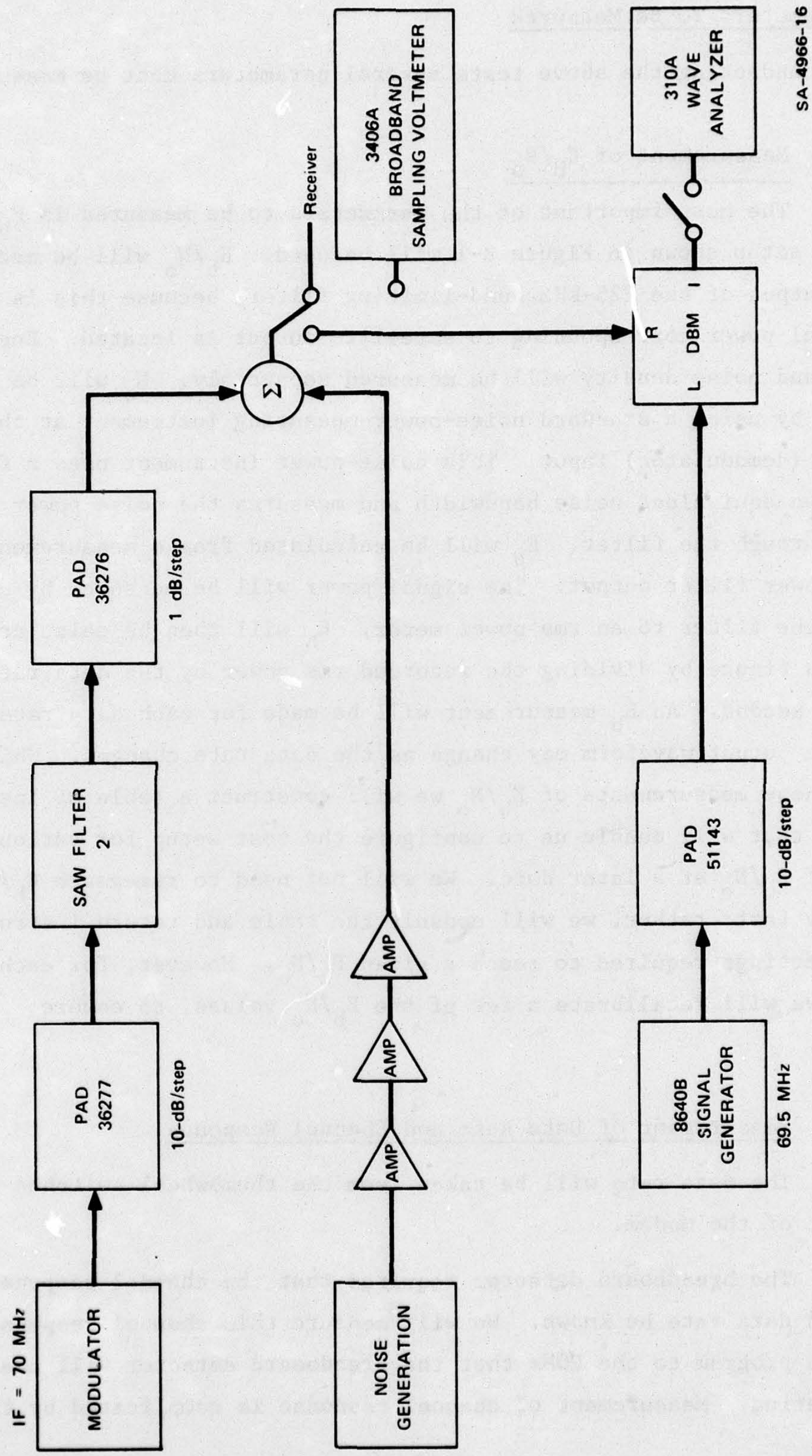


FIGURE E-1 TEST SETUP FOR MEASURING E_b/N_o

fact that a 3-bit quantizer is incorporated in the modem. Consequently, the overall channel, including the modem, is nonlinear, and a simple linear measure of the pulse response cannot be made. Therefore, we will measure pulse response in the presence of input noise by cross-correlating a known input sequence with the output sequence. This cross correlation will give a measurement of the pulse response. The known input to the modem that will be used in this cross correlation and measurement of channel response is the 2047-bit sequence of pseudo-random bits that is internal to the modem. The modem output data will be taken by a computer. After the computer has stored the input and output sequences, it will also be used for calculation of the cross correlations.

c. Error Rate

The bit-error rate will be measured by using the internal comparator in the modem. The internal comparator has an output jack on the modem front panel, to which a counter will be connected.

d. Check of Modem Implementation

The performance of the modem itself in the presence of heavy intersymbol interference but without thermal noise will be measured by setting up a back-to-back test of the modem. The internal pseudo-random pattern of the modem will generate input bits. We will display the modem output on an oscilloscope and observe the output to see whether it is stable at high data rates. In this test actual errors in the output will not concern us, because they can be caused by intersymbol interference. However, since the filter is constant and the pseudo-random pattern is repetitive, the patterns of errors and output bits should remain constant. Pattern variations could indicate modem implementation problems.

e. Statistics on Soft Decisions

Statistics on soft decisions output by the breadboard will be obtained by using the computer as we will have done to measure the channel response. Both the channel input and the breadboard output will be measured, and from these two measurements a conditional distribution of soft decision output as a function of channel input will be constructed.

5. Performance of the Tools

a. Baseline Modem Performance

The performance of the modem itself will be measured by using the experimental setup shown in Figure E-2. The error rate of the modem will be measured for each of the data rates and each of the values of E_b/N_o shown below.

Data Rate (kbits/s)		E_b/N_o (dB)
112		6
225		8
281	25 pairs	10
337		12
450		14

In these tests the input bits to the modem will be the 2047-bit sequence of pseudo-random bits. The settings on the attenuators required to realize the values of E_b/N_o will be read from the table constructed when the E_b/N_o measurements were done earlier. A few calibrating measurements will be made.

b. Test of Modem Bit Synchronization and Carrier Recovery in the Presence of Heavy Intersymbol Interference

This test setup will also be according to Figure E-2. The modem outputs called sign bit, most significant bit, and least significant bit will be examined for stability and repeatability at the above data rates. E_b/N_o will be set to a very high value, so that the only degradation will come from the modem problems with intersymbol interference.

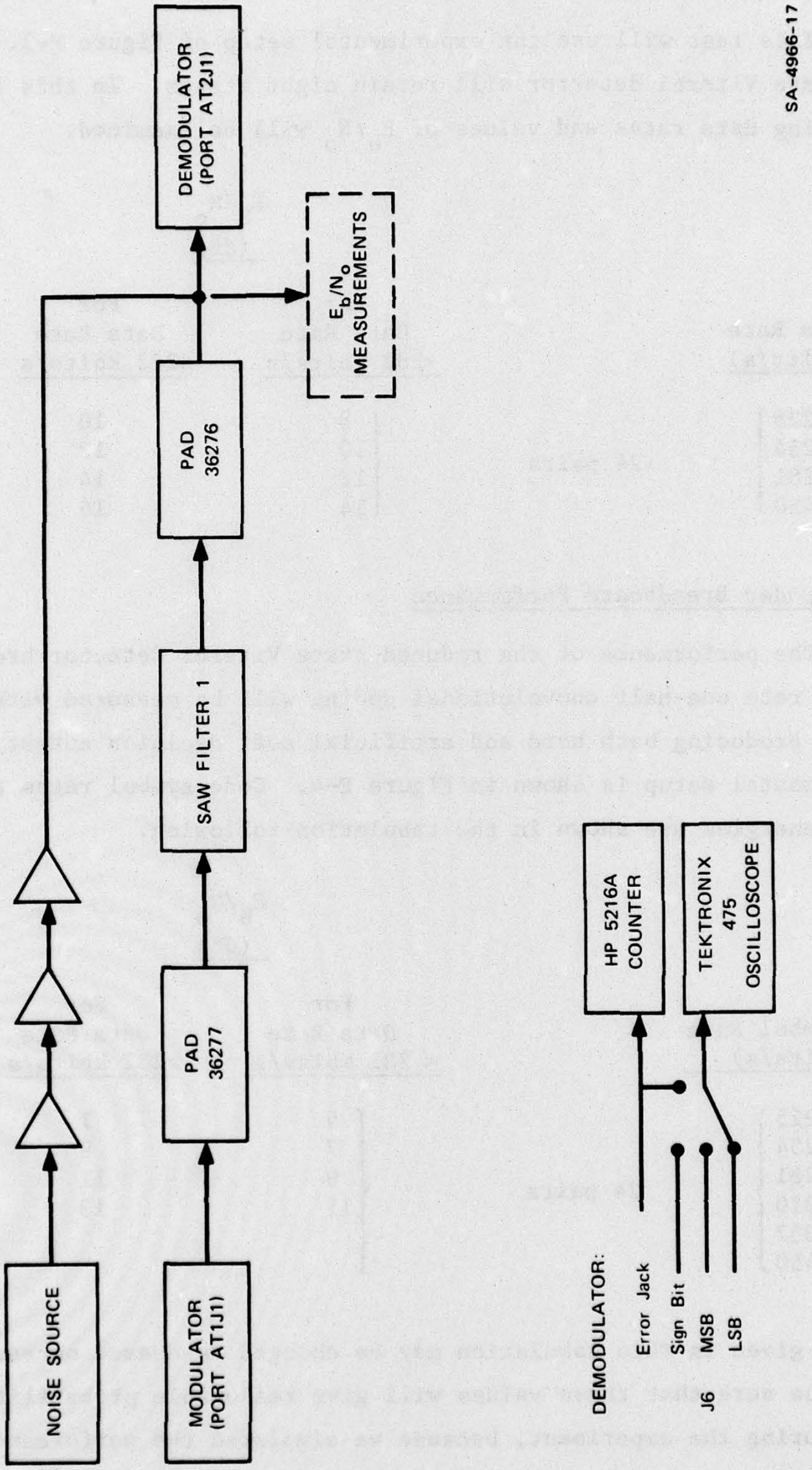


FIGURE E-2 TEST SETUP FOR MEASURING MODEM PERFORMANCE

c. Uncoded Breadboard Performance

This test will use the experimental setup of Figure E-3. The reduced-state Viterbi detector will retain eight states. In this test the following data rates and values of E_b/N_o will be examined.

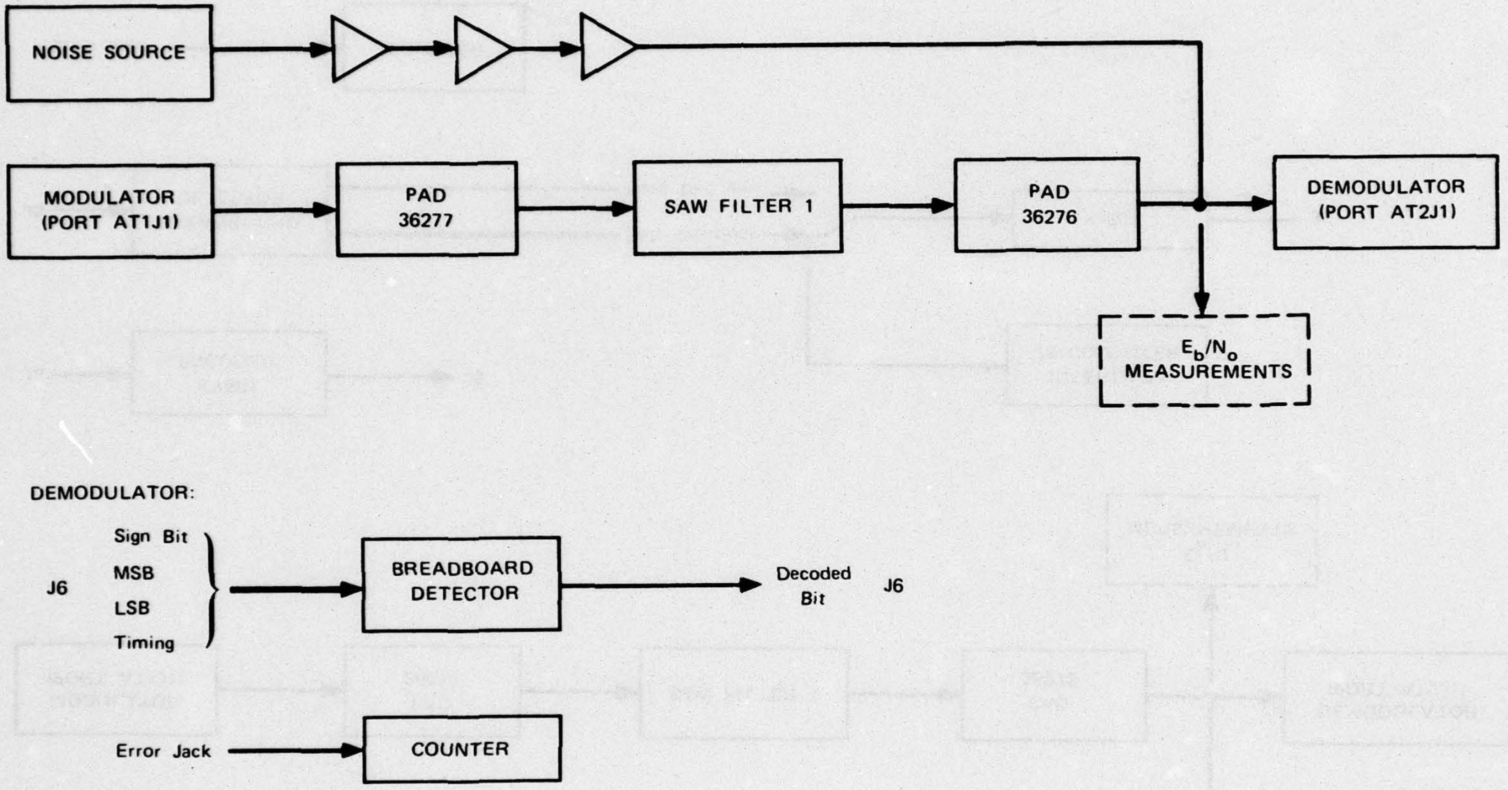
		E_b/N_o (dB)	
Data Rate (kbits/s)		For Data Rate <282 kbits/s	For Data Rate >282 kbits/s
225	24 pairs	8	10
254		10	12
281		12	14
450		14	16

d. Coded Breadboard Performance

The performance of the reduced-state Viterbi detector breadboard with rate one-half convolutional coding will be measured with the breadboard producing both hard and artificial soft decision output. The experimental setup is shown in Figure E-4. Code symbol rates and code chip energies are shown in the tabulation following.

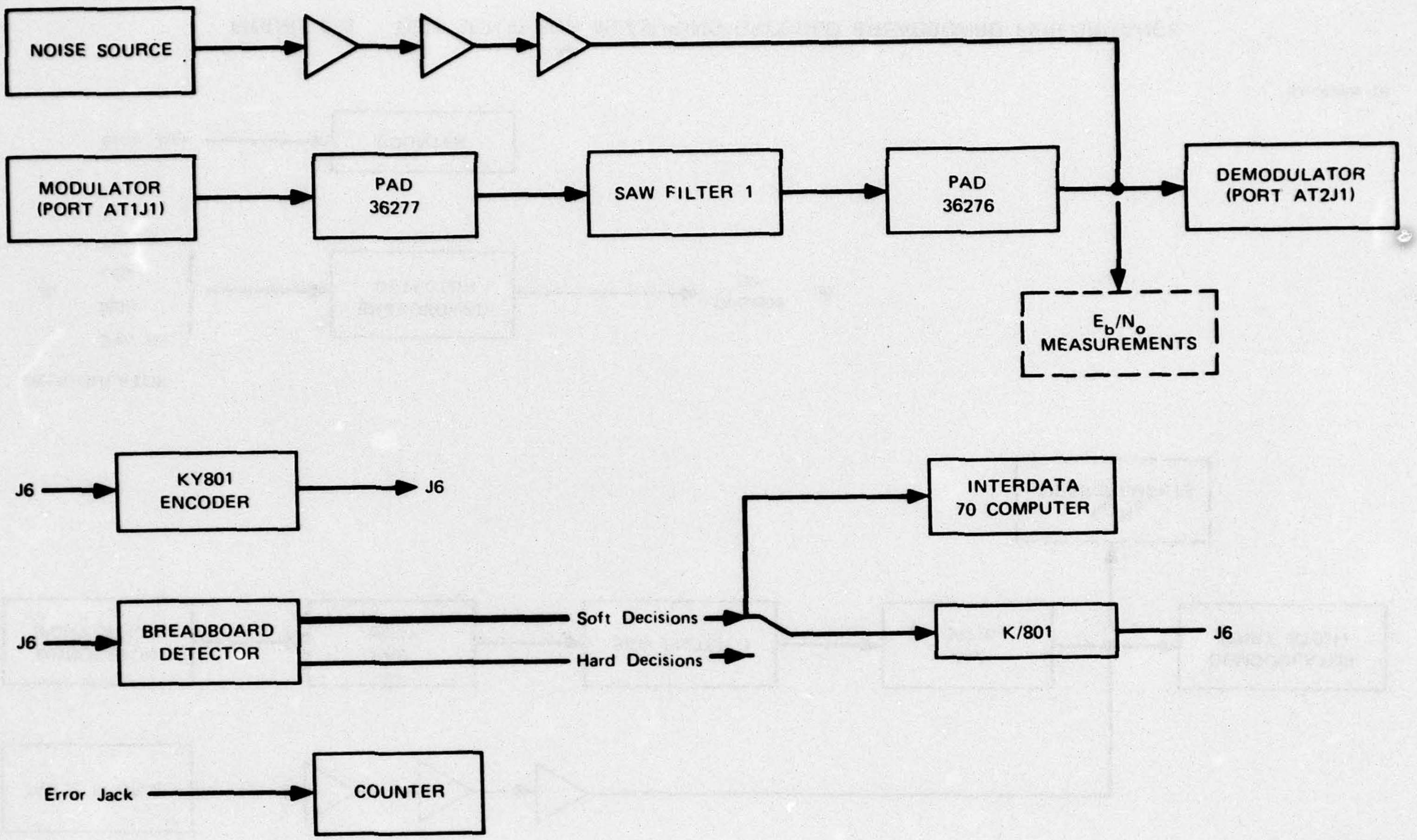
		E_b/N_o (dB)	
Code Symbol Rate (kbits/s)		For Data Rate < 282 kbits/s	For Data Rate >282 kbits/s
225	24 pairs	5	7
254		7	9
281		9	11
310		11	13
337			
450			

The values given in this tabulation may be changed as deemed necessary. We cannot be sure that these values will give reasonable probabilities of error during the experiment, because we simulated the performance



SA-4966-18

FIGURE E-3 TEST SETUP FOR MEASURING UNCODED BREADBOARD PERFORMANCE



SA-4966-19

FIGURE E-4 TEST SETUP FOR MEASURING CODED BREADBOARD PERFORMANCE of the breadboard detector with a Viterbi decoder only in the presence of interleaving. The experimental setup shown in Figure E-4 does not include an interleaver; consequently, poorer performance is expected.

The bit error probability will be measured for both hard decisions and soft decisions, using the above data. At the same time that the bit error probability measurements are being made for soft decisions, the statistics on the conditional probability distribution of soft decisions, given impact bits, will be obtained by the computer.

Preceding Page Blank - Filmed

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)

(6) Reduced State Viterbi Detector Demonstration .

5. TYPE OF REPORT & PERIOD COVERED

(9) Final rept.
Feb[redacted] 1976-[redacted] Apr 77,
April 1977

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)

(10) S. P./Russell, S. E./Cannon, & A./Shohara

8. CONTRACT OR GRANT NUMBER(s)

(15) DCA100-76-C-0037 New

9. PERFORMING ORGANIZATION NAME AND ADDRESS

Stanford Research Institute ✓
333 Ravenswood Avenue
Menlo Park, California 94025

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS

Defense Communications Engineering Center, DCA
1860 Wiehle Avenue
Reston, Virginia 22090

12. REPORT DATE

(11) Apr[redacted]977

13. NO. OF PAGES

103

15. SECURITY CLASS. (of this report)

Unclassified (12) 78p.

14. MONITORING AGENCY NAME & ADDRESS (if diff. from Controlling Office)

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this report)

Approved for Public Release--Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Communication satellite, band-limiting, Viterbi, intersymbol interference, digital modulation.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Early work on a new detection method applicable to band-limited satellite channels was extended. The method, termed reduced-state Viterbi detection, is a derivative of the Viterbi algorithm. In the study reported here, the potential of combining

332500

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

19. KEY WORDS (Continued)

20 ABSTRACT (Continued)

Viterbi detection of band-limited sequences with Viterbi decoding of error-correctio codes was examined through simulation. A breadboard reduced-state detector was designed, and results were obtained from actual test. These results verified the predictions of the earlier study and showed that reduced-state detection can lead to significant increases in band-limited satellite channel performance.